EXHIBIT 2

IN THE UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF TEXAS
MIDLAND/ODESSA DIVISION

VIRTAMOVE, CORP., PLAINTIFF, v. AMAZON.COM, INC.; AMAZON.COM SERVICES LLC; AND AMAZON WEB SERVICES, INC., DEFENDANTS.	CASE NO. 7:24-CV-00030

VIDEOTAPED DEPOSITION OF DONN ROCHETTE
TAKEN REMOTELY VIA ZOOM VIDEOCONFERENCE
TUESDAY, SEPTEMBER 10, 2024
11:04 A.M. CDT

REPORTED BY AUDRA E. CRAMER, CSR NO. 9901

DIGITAL EVIDENCE GROUP
1730 M Street, NW, Suite 812
Washington, D.C. 20036
(202) 232-0646

VIDEOTAPED DEPOSITION OF DONN ROCHETTE, TAKEN REMOTELY VIA ZOOM ON BEHALF OF THE DEFENDANTS AT 11:04 A.M. CDT, TUESDAY, SEPTEMBER 10, 2024, BEFORE AUDRA E. CRAMER, CSR NO. 9901, PURSUANT TO SUBPOENA.

APPEARANCES OF COUNSEL

ON BEHALF OF THE PLAINTIFF:

RUSS AUGUST & KABAT
BY: PETER TONG, ESQUIRE
4925 GREENVILLE AVENUE, SUITE 200
DALLAS, TEXAS 75206
(310) 826-7474
ptong@raklaw.com

ON BEHALF OF THE DEFENDANTS:

KNOBBE, MARTENS, OLSON & BEAR LLP
BY: JEREMY ANAPOL, ESQUIRE
2040 MAIN STREET, 14TH FLOOR
IRVINE, CALIFORNIA 92614
(949) 760-0404
jeremy.anapol@knobbe.com

ALSO PRESENT

BILLY FAHNERT, VIDEOGRAPHER

I N D E X

E X H I B I T S

NO.	PAGE	DESCRIPTION
1010	14	US PATENT 7,519,814
1009	18	ARCHIVE VERSION OF THE HOMEPAGE ONCORE SYSTEMS CORPORATION
1011	24	US PATENT 7,784,058
1012	42	AUSTRALIAN UNIX SYSTEMS USER GROUP NEWSLETTER, VOLUME 8, NUMBER 5

QUESTIONS INSTRUCTED BY COUNSEL NOT TO ANSWER

PAGE	LINE
144	7
145	2

REPORTER'S NOTE: All quotations from exhibits are reflected in the manner in which they were read into the record and do not necessarily indicate an exact quote from the document.

REMOTELY VIA ZOOM VIDEOCONFERENCE

TUESDAY, SEPTEMBER 10, 2024, 11:04 A.M. CDT

THE VIDEOGRAPHER: We are on the record. This is the remote video deposition of Donn Rochette in the matter of VirtaMove Corp. versus Amazon.com, Inc., et al., filed in the United States District Court for the Western District of Texas.

My name is Billy Fahnert. I am the video technician today. The court reporter is Audra Cramer. We both represent Digital Evidence Group.

Today's date is September 10, 2024. The time is 11:04 a.m. Central Standard Time.

All parties have stipulated to the witness being sworn in remotely.

Will Counsel please identify yourselves for the record, and then the witness will be sworn in.

MR. TONG: This is Peter Tong from Russ August & Kabat on behalf of VirtaMove

Corporation?

MR. ANAPOL: Jeremy Anapol of Knobbe Martens Olson & Bear on behalf of Defendants.

DONN ROCHETTE,

having been first duly sworn, was examined and testified as follows:

MR. ANAPOL: Thank you.

EXAMINATION

BY MR. ANAPOL:

Q. Mr. Rochette, thank you again for being here today, taking time out of your schedule to help us collect some information that we hope will be helpful in our case.

Have you ever been deposed before?

A. No.

Q. Okay. So I'll cover some general background with you just to help you understand the process.

But before that, can we start by having

you state your full name for the record?

A. My name is Donn Rochette.

Q. And Mr. Rochette, where are you joining us from today?

A. I am calling from my home in northern Iowa.

Q. Okay. So the deposition is going to follow a question-and-answer format, and Audra, our court reporter, is going to be transcribing the answers. Because she can only transcribe one person speaking at a time, I'll ask that you wait until I finish my question to start answering, and I will try to wait until you finish answering to ask the next question.

Mr. Tong, who's representing the Plaintiffs, may have some objections which he might chime in with, but you can go ahead and answer the question after he finishes his objection. Those are for the record, so, if necessary, the judge will look at those objections later, but you don't need to concern yourself with those.

A. Okay.

Q. If you don't understand one of my questions, please ask for clarification. I'm happy to provide it.

If you need to take a break at any time, please let me know, but if there's a question that's pending, I'll just ask that you give the answer, and then we'll take a break afterward.

A. Okay.

Q. Is there any reason why you cannot provide complete and accurate testimony today such as, you know, being on any medication or anything like that?

A. No.

Q. And, Mr. Rochette, you used to work at a company called Trigence Corp.; is that right?

A. Yes.

Q. And do you remember when you worked for Trigence?

A. Gosh, I couldn't give you the dates off the top of my head, no. It's been a long time.

Q. Was it in the early 2000s?
A. Yes.
Q. And did you understand that Trigence later became AppZero?
A. Yes.
Q. And then there's a company called AppZero Software Corp. which is now called VirtaMove.
Do you understand that?
A. I was not aware of that, no. Not until very recently.
Q. And when did you become aware of that?
A. Oh, about two months ago someone from VirtaMove reached out to me over LinkedIn --
Q. Okay.
A. -- and pointed out the connection.
Q. Do you remember who from VirtaMove reached out to you over LinkedIn?
A. The person identified himself as the CEO of VirtaMove. I can't remember his name. Sorry.
Q. Was his name Nigel Stokes?

A. That sounds familiar, yes.
Q. And what did you talk about with Mr. Stokes?
MR. TONG: Objection. Foundation.
MR. ANAPOL: I'll rephrase the question.
Q. So occasionally, Mr. Rochette, if Peter makes an objection like that, I might rephrase the question to help resolve the objection.
After this person purporting to be Mr. Stokes reached out to you via LinkedIn, did you speak with him?
A. Yes, I did.
Q. And he identified himself as Mr. Stokes when you spoke to him?
A. Yes.
Q. Okay. And what did Mr. -- what did you talk about with Mr. Stokes?
A. He identified the fact -- he explained the fact that there was a lawsuit being formed over a couple patents that I had done. Basically asked me if I wanted to get involved.

I said in no way was I interested in getting involved. And that was it.

Q. And did you tell him why you didn't want to get involved?

A. Yes. I said I'm retired. I don't want to have anything do with it. I don't have an opinion one way or the other. I don't have any involvement in it, and I would prefer to not be involved.

Q. And when you say you don't have an opinion about, what was it that you didn't have an opinion about?

A. The fact that the company called VirtaMove was suing several companies regarding a couple of the patents that I did, and I don't have any -- yeah, I -- what am I trying to say? I don't have any opinion one way or the other about that.

Q. You mean opinion about the merits of the lawsuit?

A. Well, it seemed like something that was far-reaching, that was not very -- didn't make a

whole lot of sense, and didn't see where it would go anywhere, didn't understand my involvement, how I could help at all in it.

And so that's the reason why I said to him I don't want anything to do with it. It just seemed rather bizarre and farfetched, and I couldn't appreciate the merit of it at all.

Q. And did Mr. Stokes suggest a particular role for you in the lawsuit?

A. No. He only suggested that I -- would I speak with counsel on his part, and I declined.

Q. Did he offer you any sort of compensation?

A. No.

Q. So have you spoken with anyone else at VirtaMove besides Mr. Stokes?

A. No.

Q. And it was just that one conversation that you had with him; correct?

A. That is correct.

Q. So what was your role when you worked

at Trigence?

A. I was the chief technology officer and a software developer.

Q. And can you explain what you were responsible for in those roles.

A. I was responsible for the overall design and implementation of the software we were building, of the architecture of that software and the direction of it.

Q. And what was the software you were building while you were at Trigence?

A. Trigence was focused on software for the corporate data center, for applications specifically running for corporate purposes in, for the most part, large data centers.

Q. And what did the software do?

A. It would allow multiple versions of an application to run on one instance of an operating system.

Q. And what was the name of that software?

A. Good question. I don't remember. App something, I think.

Q. Was it Trigence AE?
A. Oh, yeah. That's it. Yes. Thank you.
Q. And --
A. Yeah. Yes.
Q. And did the "AE" stand for "application environment"?
A. Yes, I did.
Q. And you worked on several patents while you were at Trigence; correct?
A. Yes.
Q. And do you understand that two of those patents are asserted in this case?
A. Yes.
Q. And I'm just going to show you some documents to help with this discussion.
A. Okay.
MR. ANAPOL: And so I'm going to ask Billy to mark them, but first I have to put them where Billy can see them.
So, Billy, could you go ahead and mark Document A as Exhibit 1010, please. 1010.
THE VIDEOGRAPHER: Okay. Give me one

moment to pull it down.

(Whereupon, Exhibit 1010 was marked for identification.)

BY MR. ANAPOL:

Q. And, Mr. Rochette, do you recognize Exhibit 1010?

A. Yes.

Q. And this is a copy of US Patent No. 7,519,814.

Do you see that --

A. Yes.

Q. -- in the upper right?

A. Yeah.

Q. And you are listed as the first named inventor on this patent.

Do you see that?

A. Yes.

Q. So where it says "Donn Rochette, Fenton, Iowa, United States," that's you; correct?

A. That is me, correct.

Q. And does the '814 patent generally

relate to containers?

A. Yes.

Q. When did you first become aware of containers?

A. Early 2000s. Would have been about a year -- it would have been about two years before this patent was published. So, yeah, we were discussing the concepts and examining ways of creating an environment where applications can run -- disparate applications could run, yes.

Q. And I just want to clarify one thing you said. You said "about two years before this patent was published."

Did you mean published or filed?

A. Filed. Thank you for the clarification. Yes.

Q. So you can see on this page here, on the bottom left corner of the portion that Billy was showing -- Billy, if you scroll up.

Under "Related US Application Data," it refers to two provisional applications that were

filed in September 2003.

A. Uh-huh. Right.

Q. So those are the initial disclosures that this is based on.

Is that the filing date that you were referring to --

A. Yes.

Q. -- when you were saying two years earlier?

A. Yes. We would have been discussing and talking about and envisioning things, yes, before this was -- before the provisional was filed.

Q. And how did you first learn about containers?

MR. TONG: Objection. Foundation.

BY MR. ANAPOL:

Q. You can go ahead and answer, Mr. Rochette.

A. Oh, okay.

I first learned about containers in discussions with corporate -- with people

running corporate data centers and their description of how difficult it was to get applications to run, to migrate older applications to newer platforms, the number of servers and hardware instances that were required for each application, things like that.

And so we began envisioning an environment where multiple applications could run and trying to determine if we could solve the problem where an application that was run on one version of the operating system could run on a newer version of a very similar operating system was what -- some of the first things that caused us to think to arrive at what is now described here as containers.

Q. Okay. But you worked on containers at your prior employer before Trigence; right?

MR. TONG: Objection. Foundation.

THE WITNESS: Yes, they would very loosely be called containers in a prior job.

The work I did prior to Trigence was focused on embedded systems. So these were --

this is software that would run on a computer platform that in itself is a part of a larger product, so the engine controller in your car or, you know, some sort of controller for a larger purpose, the controlling software that would be a part of that embedded computer is what we were working on.

So we didn't at the time think of them as containers, but you could loosely define them as containers in some respect because they are certain applications running, albeit for a lightly different purpose.

MR. ANAPOL: Billy, could you mark Document C as Exhibit 1009.

(Whereupon, Exhibit 1009 was marked for identification.)

BY MR. ANAPOL:

Q. So, Mr. Rochette, Exhibit 1009 is what appears to be an archive version of the homepage for OnCore Systems Corporation.

Does that -- do you recognize this?

A. Yes. That is a blast from the past.

Q. And is OnCore Systems Corporation where you worked before Trigence?

A. Yes.

Q. And do you see in the second black paragraph on this page what's written in bold?

A. Yeah, you're right. They were called virtual computing containers, yeah.

Q. So OnCore, where you worked before Trigence, had a capability that they called virtual computing containers; right?

A. That is correct.

Q. And the OnCore operating system allowed users to run Linux applications; correct?

A. Yes.

MR. TONG: Hold on. Before your next question, I want to object to this whole line of questioning as exceeding the scope of venue discovery that you're entitled to ask Mr. Rochette about. Venue discovery should be limited to the locations of witnesses, the relevance of evidence and the inconvenience cost of trial, and we're getting into fact discovery.

BY MR. ANAPOL:

Q. Mr. Rochette OnCore Systems Corporation was located in California; correct?

A. Yes. The corporate headquarters were located in California, yes.

Q. And so OnCore Systems Corporation was developing containers in California; right?

A. Yes.

Q. In Northern California specifically; correct?

A. Yes. In the Bay Area.

Q. Thank you.

And the OnCore operating system, which you helped develop before you went to Trigence, allowed users to run Linux operations; correct?

A. Correct.

Q. And OnCore called its system for walling off applications virtual computing containers; correct?

A. Yes.

Q. And OnCore had those virtual computing containers before Trigence developed its

container product; correct?

A. Correct. Yes.

Q. And when you worked for OnCore, were you physically in Northern California?

A. No. I worked out of Iowa, near Fenton.

Q. Did you visit Northern California for work?

A. I did, yes.

Q. How often did you visit Northern California?

A. Probably three or four times a year.

Q. And how long would you stay when you visited?

A. Most often not more than a week at a time.

Q. And did you have contact during your time at OnCore with software developers who were physically in Northern California?

A. Yes.

Q. Did OnCore have any employees who were employed by OnCore that were physically based in Northern California?

A. Yes.
Q. And did you speak with those people when you would visit Northern California?
A. Yes.
Q. And what was your title at OnCore?
A. The chief technology officer.
Q. So were you well acquainted with how OnCore's operating system worked?
A. Yes.
Q. And were you well acquainted with how OnCore's containers worked?
A. Yes.
Q. And before Encore, did you work for a different company in Northern California?
A. Yes. I worked for Microtec -- well, wait. Microtec was bought by another company out of Oregon. I can't remember the name of it.
Yes, we lived and worked in Northern California before OnCore.
Q. So when did you live in Northern California?
A. Early 2000s.

Q. Okay. And when did you stop living in Northern California?

A. Oh, I think -- I don't know. 2001, 2002, something like that, we moved to Iowa. I'd have to go back and check the exact dates.

Q. Is that around the same time that you started working at Trigence?

A. It's around the same time that I started working at OnCore. I moved to Iowa, working remotely for the company based in California, and then shortly thereafter I started working for OnCore in Iowa.

Q. Okay. And then after OnCore you started working at Trigence; correct?

A. Correct. Yes.

Q. And did you remain in Iowa when you worked for Trigence?

A. Yes.

Q. And then briefly, what did you do at Microtec, the company you worked at before OnCore?

A. I was a software developer working on

real-time operating systems.

Q. So before Trigence, how long would you say you worked in the operating system field?

A. I think it was around 1998 or 1999 I started focusing specifically on real-time operating systems.

Q. Okay. So several years before you filed the application for the '814 patent, you were a software developer in Northern California working on operating systems before you moved to Iowa; correct?

A. That is correct.

MR. ANAPOL: Billy, can we mark Exhibit B as Exhibit 1011.

(Whereupon, Exhibit 1011 was marked for identification.)

BY MR. ANAPOL:

Q. Mr. Rochette, do you recognize Exhibit 1011?

A. Yes.

Q. This is a copy of US Patent No. 7,784,058.

Do you see that?
A. Yes.
Q. And you are listed as the first named inventor on this patent; correct?
A. Yes.
Q. And just for the record, I will refer to this as the '058 patent. Okay?
A. Okay.
Q. And Exhibit 1010, which we looked at before, I will refer to as the '814 patent. Okay?
A. Okay.
Q. And the '058 patent generally relates to shared libraries; correct?
A. Yes.
Q. What is a shared library?
A. It is a grouping of code that is assembled into an archive that gets linked into an application that would be called a binary or an executable.
So almost any executable in any of the operating systems that you would be familiar

with would have several shared libraries linked to it. In Unix they're called shared libraries. In the Windows world they're called DLLs, or dynamic link libraries.

Q. So the Windows equivalent of a Unix shared library is a dynamic link library?

A. Correct.

Q. And when did you first become aware of shared libraries?

A. Very early on in my career. They have been a part of Unix and Linux and Windows for -- since the beginning, so I would have been aware of them very early in my career.

Q. By the 1980s would you have been aware of shared libraries?

A. Yes, definitely.

Q. And do you understand that -- let me withdraw the question and rephrase.

Are you familiar with a company called sun Mike systems?

A. Yes.

Q. And Sun Microsystems was based in

Northern California; correct?

A. Yes.

Q. And you were aware of an operating system from Sun Microsystems called Solaris?

A. Yes.

Q. And Solaris predated the two patents that we're looking at; correct?

A. That's correct.

MR. TONG: Objection. Vague.

THE WITNESS: Sorry.

That is correct.

BY MR. ANAPOL:

Q. Okay. And before -- long before you filed these patent applications, Solaris had support for shared libraries; correct?

A. Yes.

Q. And you were aware of Solaris's support for shared libraries before you filed the applications that led to these two patents; correct?

A. That is correct.

Q. And many other operating systems also

had shared libraries; correct?

A. Correct.

Q. And are you aware of any examples, from your personal experience, of developers moving the functionality from an operating system's kernel to a shared library?

A. There is -- there are some examples. You have to -- you have to -- it's not as clear-cut as move code literally from the kernel into a shared library, but functionality that you would normally expect to find in the kernel was -- has been -- had been implemented in libraries.

Q. Can you give me an example of that?

A. [Garbled] zones was a really big example of it.

(The reporter requested clarification.)

THE WITNESS: A feature in the Sun Microsystems operating system called Solaris, the feature is callid zones, z-o-n-e-s.

BY MR. ANAPOL:

Q. And do you know where the people

responsible for Solaris zones would be located?

A. The Bay Area --

MR. TONG: Objection. Vague.

THE WITNESS: Oh, sorry.

BY MR. ANAPOL:

Q. Go ahead, Mr. Rochette. Sorry for the interruption.

A. Yeah, they're located in the Bay Area of California, yes.

There's also examples where Apple has done the same kind of thing, and it's not -- I'm sorry. I'm answering questions you didn't ask. Sorry.

Q. Well, I did ask for examples.

So it sounds like you were going to mention Apple as another example; is that right?

A. Yeah, Apple is another example with the -- with the diversion of macOS that we currently know, there's cases where Apple has moved things that would be expected to be in other operating systems. Like, for example, Linux would be in the kernel or not not in the

kernel in the macOS example.

Q. And are you referring to the macOS version that is based on Unix?

A. Yes.

Q. And that is OS X?

A. Correct.

Q. And do you know where Apple is headquartered?

A. Cupertino, California, in the Bay Area.

I met with groups of both of these developers.

Q. When did you meet with these groups of developers?

A. Early on in the Trigence days.

I'm sorry. No. Wait. Excuse me. I met with Apple at the end of the OnCore days. I met with the Sunday's Solaris people early on in the Trigence days. So it's split between the two companies.

Q. All right. So you met with the Apple folks before you started working at Trigence?

A. That is correct.

Q. And what was that discussion with the Apple folks about?

A. I met with the chief technology of Apple at the time. His name is Avie Tevanian. And Avie was deep into doing very similar things that we were doing at OnCore with a microkernel and multiple-user-mode operating system elements running, including shared libraries.

And we were working with a microprocessor architecture called the PowerPC. Avie and his teams were working with the microprocessor architecture with -- I think they moved to Intel at the time. But anyway, they were not -- and so they were interested in our experience with that hardware architecture, and we were discussing lots of deep software things related to CPU architecture and this mode of operation.

Q. Can you repeat the last name of this Avie.

A. Avie Tevanian. I can't remember how to spell it.

Q. That's fine. We can look it up. Thank you.

So did Mr. Tevanian reach out to you, or did you reach out to him? Do you recall?

A. My recollection is that one of the salespeople that worked for OnCore in Northern California had a conversation that led to an interaction with Avie Tevanian, which led to an invitation for me to come out and meet with him.

Q. And where did you meet with him?

A. In Cupertino in his office.

Q. And at the time you met with him, had Apple started incorporating this functionality that would normally be in the kernel into a shared library --

A. Yes.

Q. -- or was that a new feature?

A. Yes. No, they were well underway with that, yes.

Q. And do you remember anything else about what you discussed with Mr. Tevanian?

A. No, nothing -- he wanted us to do some

work for them, and we -- and that was about -- that's all I can recall. Lots of low-level technical detail.

Q. Low-level technical detail related to operating system implementations?

A. Yes.

Q. And how long did you spend talking with Mr. Tevanian? Do you remember?

A. My recollection would be a couple of hours.

Q. Was it just, like, one session, or were there multiple sessions?

A. There was only one session face-to-face.

Q. Were there some phone calls or --

A. Email exchanges. There were email exchanges.

Q. All right. And then what about your interactions with Solaris folks? Do you remember who you spoke to at Sun Microsystems?

A. No, I don't remember names. I remember we met with that entire software group

responsible for Solaris, and I don't remember -- I don't remember the -- which -- the name of the group. It was quite a few software developers at their headquarters and -- yeah.

Q. Do you remember if this was before or after 2003?

A. No. I only remember it being early in the Trigence days. I don't remember a date.

Q. Do you remember if it was before or after you filed these patent applications?

A. I don't remember for sure, no. I only have a feeling.

Q. What is your feeling?

A. I think that the provisionals had been filed, but I can't say for sure.

Q. Okay. Can you tell me if you recognize any of these names: Andrew Tucker, John Beck, David Comay, Andrew Gabriel, Ozgur Leonard or Daniel Price?

A. I do recognize two of them.

Q. Which two?

A. John Beck and -- can you read the list

again, please.

Q. Sure.

Andrew Tucker, David Comay, Andrew Gabriel, Ozgur Leonard, Daniel Price.

A. I recognize David's name, David Comay, and John Beck.

Q. Do you know if these were among the people you spoke to at Sun Microsystems?

A. I can't remember, no.

Q. How do you know Mr. Beck?

A. We met with him. I had conversations with him. I remember -- I just remember the name, because I -- when I worked with NASA in Florida, there was -- I worked with a guy by the name of John Beck, and I remember, you know, hey, another John Beck. You know, that just -- just a memory trigger is all.

Q. Okay. But the John Beck -- let me clarify.

Do you remember a John Beck who was affiliated with Sun Microsystems?

A. Yes.

Q. And what do you remember about David Comay?

A. The name is familiar. I think it was in the meetings we had with Sun. That's -- I don't remember. I can't -- I don't remember why his name is familiar, but I recognize it.

Q. Okay. And what did you discuss in your meetings with the Sun Microsystems folks?

A. We discussed the differences between Solaris zones and what we were doing at Trigence.

Q. And what was the difference?

A. Well, there's a lot of differences. The guys at Solaris were concerned that -- it was their feeling that we could not have implemented what we were doing in Solaris without having access to Solaris source code, and we not had any -- Solaris is closed source and was -- we didn't have any access to it.

And it was somewhat confrontational, and we had to sit down and explain to them how we were doing what we were doing and the way in

which it did not require internal knowledge -- it didn't require knowledge to the internals of Solaris in order to do what we were doing. And they -- in the end -- it was a rather lengthy discussion. In the end they recognized what we were doing, and we parted friends. Everything was fine.

Q. Okay. So initially Trigence AE was for Linux; right?

A. Correct.

Q. And --

A. Linux and Windows.

Q. Linux and Windows?

A. Yes.

Q. And then Solaris support was added later?

A. Yes.

Q. Okay. And so the meeting with Sun Microsystems was after you had developed some sort of Solaris capability for Trigence AE?

A. I don't think it was released to the public, but, yes, we had working versions. Yes.

Q. Okay. So what your understanding of Solaris zones?

A. Oh gosh. That is a lengthy answer.

Zones would provide a separate container, a separate security environment for each app and for our multiple -- and for an application to run it. So you could define a zone for each application you had running in Solaris. That zone would allow the application to have its -- potentially its own version of files. It would -- I don't recall if it had isolated networking, but it would provide a secure environment, and it would provide a separate instance of files and so forth.

And, it was -- if you look at it at a very high level, it looked somewhat like what Trigence AE was doing very differently and at different -- very different results but largely -- largely the same -- well -- anyway, at the very, very highest level, objectives were similar.

Q. Okay.

A. Does that help?
Q. Sure.
So what are some of the differences between what Trigence AE was doing and what Solaris zones was doing?
A. With zones, Solaris would utilize some operating system features to create a separate file area. They had modified the kernel, the internals of Solaris, to support this. It relied on kernel internals to provide separations between applications' end zones. It would provide separate scheduling types of mechanisms and so forth also implemented in the kernel.
The Trigence solution made no changes to the kernel -- or didn't require any changes to the kernel. It was all done with what was called function overlays, where the calls made to shared libraries could be, essentially, redirected to calls in the Trigence library, and we were able to manipulate things in that way. It had nothing to do with separate zones or

kernel changes or anything. So...

Q. Okay. So when you say that Solaris zones was different from what Trigence was doing, you were contrasting Solaris zones with the Trigence product, not the patents; correct?

A. Yes.

Q. So can you explain these function overlays to me a little bit more.

A. Function overlays use an operating system capability called library preload. In Unix and Linux and Solaris it's called LD_Preload. In Windows it's called AppInit_DLL. And what happens is you would use LD_Preload or AppInit_DLL to identify a library that you wanted loaded with your -- with an application as it starts. When that application starts, there's a dynamic loader that would load the application plus other requisite libraries.

And LD_Preload/AppInit_DLL would say in addition to the requisite libraries, also load this library, and load it first before any other libraries. So that has the effect of managing

the -- what's called the namespace that the loader used.

So if the application called a function called fu, let's say, for example, if fu were existent in a system library and also in a Trigence library, because of the preload, the loader would resolve fu to the Trigence library instead of the system library because the Trigence library was loaded first due to LD_Preload/AppInit_DLL.

Q. Okay. So you mentioned that this function overlay capability that Trigence was using was called LD_Preload in Solaris; is that right?

A. Correct.

Q. Okay. So Solaris had this LD_Preload capability that could be used to provide the function overlay capability?

A. Correct.

Q. And do you know when Solaris gained that function overlay capability that Trigence was using?

A. That capability has been present in Solaris since its first release of its origin.

Q. Does the 1987 sound like approximately the right timeframe?

A. Yeah, that would be -- that would be in the range, yes.

Q. I'm going to show you a document and see if it's referring to the same thing you're talking about when you say function overlay.

A. Okay.

MR. ANAPOL: So, Billy, can you mark Document E as Exhibit 1012.

(Whereupon, Exhibit 1012 was marked for identification.)

(Discussion held off the record.)

BY MR. ANAPOL:

Q. So, Mr. Rochette, Exhibit 1015 the cover page says "Australia Unix Systems User Group Newsletter, Volume 8, No. 5."

Do you see that?

A. Yes.

MR. ANAPOL: And, Billy, can you go to

page 3 of the PDF?

Q. Mr. Rochette, do you see that this is the table of contents?

A. Yes.

Q. And do you see that it's dated October 1987?

A. Yes.

Q. And towards the bottom, corresponding with page 112, there's an entry for "Shared Libraries in SunOS."

Do you see that?

A. Yes.

MR. ANAPOL: So, Billy, could you go to page 114 of the PDF?

Q. And, Mr. Rochette, you see that this page shows the cover of an article entitled "Shared Libraries in SunOS"?

A. Yes.

Q. And you see that there's four authors from Sun Microsystems.

Do you see that?

A. I do, yes.

Q. And you see that they give an address in Mountain View, California; correct?
A. Yes.
Q. And that's Northern California; right?
A. That is correct.
MR. ANAPOL: And, Billy, could you please go to page 127 of the PDF.
Q. So, Mr. Rochette, do you see the heading for Section 7.2 of this article?
A. Interposition, yeah, uh-huh.
Q. And could you just read this to yourself and let me know when you're done.
A. Yeah, I'm done.
Q. So is the interposition being described in this 1987 article the same as the function overlay capability that you were referring to in your earlier testimony?
A. It is functionally the same, yes.
Q. And did you learn about this capability in the '80s, or did you learn about it later?
A. Oh, probably did learn about it in the late '80s, yeah.

Q. Do you remember how you learned about it?

A. I don't remember the first time I used it, no. I don't recall the scenario, no.

MR. ANAPOL: Okay. Billy, you can close this.

Q. So I want to briefly go back to OnCore, and then we'll take a break because we've been going for about an hour.

Did OnCore's operating system have a networking stack during the time you worked there?

A. Yes.

Q. And let me just clarify that when I'm asking about OnCore, I'm only ask about the time you worked there, not anything that happened after. Okay?

A. Okay.

Q. So was the networking stack for the OnCore operating system implemented inside or outside of the kernel?

A. Outside of the kernel.

Q. Do you remember how it was implemented?
A. It was implemented as a separate -- what's called a process as a part of -- because all of Unix was implemented as a separate process. So it was a part of the Unix instance, which -- where all of Unix ran outside of the kernel in user mode.
Q. So you had all of Unix running in user mode on top of the OnCore operating system; correct?
A. Yes.
Q. And so all of that was outside of the OnCore operating system's kernel; correct?
MR. TONG: Objection --
THE WITNESS: Outside of the kernel, correct.
MR. TONG: And I'm going to object to this line of questioning again as getting back into deep fact discovery, outside the scope of venue discovery, and I don't think Mr. Donn Rochette needs to answer any of these questions.
MR. ANAPOL: Well, your objection is

noted.

Q. But you still have to answer the question, Mr. Rochette. And, again, I'll just -- let me just repeat the question here.

So Unix was running outside of the kernel in the OnCore operating system; correct?

A. Yes.

Q. And that includes the Unix networking stack; correct?

A. Yes.

Q. And was any part of that in a shared library?

A. Yes, shared libraries were utilized for portions of it. Yes.

Q. And did OnCore's operating system have a file system?

A. Yes.

Q. And was the file system implemented inside or outside of the operating system kernel?

A. Outside of the kernel.

Q. Was any of it in a shared library?

A. Yes, portions of it would have been implemented in a shared library -- one or more shared libraries, yes.

Q. And are there anybody -- let me rephrase.

Are there anybody -- I can't get my tenses to line up.

Are there any people in Northern California who know about that functionality that OnCore implemented in shared libraries?

A. That OnCore implemented? Well, there are numerous people in Northern California that would be familiar with the way in which it was done or the way in which it is done, yes. Specifically OnCore, I don't -- I doubt it. I don't know.

Q. Who aside from you would know, from working at OnCore, how those file system functionalities were implemented?

A. Well, just about any software developer at Apple.

Q. Okay.

A. Because Apple's architecture runs in the same way, uses the same kernel we were using.

Q. And who aside from you would know, from working at OnCore, how the networking stack was implemented?

A. Same thing. Lots of people would understand how it works, maybe not specifically what OnCore was doing, but the architecture, the concept, the techniques would all be well known by anybody at Apple.

Q. And do you know if any software developers who worked with you at OnCore are still in Northern California?

A. There's a gentleman called George Marrow that's still in Northern California. That's the only one that I know of that I can think of anyway right now.

MR. ANAPOL: Mr. Rochette, would you like to take a five-minute break? We've been going for about an hour.

THE WITNESS: I would. Thank you.

MR. ANAPOL: Back in five minutes everybody?

MR. TONG: That's fine with us.

THE VIDEOGRAPHER: Going off the record. The time is 11:59.

(Recess taken.)

THE VIDEOGRAPHER: We are back on the record. The time is 12:09.

BY MR. ANAPOL:

Q. Mr. Rochette, before the break you mentioned an individual named George Marrow.

A. Uh-huh.

Q. Could you give us the spelling of that if you know it.

A. M-a-r-r-o-w.

Q. And --

A. I haven't spoken to George in a long time. You asked if there might be anybody, and that's the only one I could think of who might be around.

Q. Okay. We can check. Thank you.

At the time that you had it these

discussions with Sun Microsystems that you testified about earlier, do you know if they had created zones yet at the time.

A. Yes, they had. It was very new, but they were out and released in the product.

Q. I just want to at this point get an overview based on your experience working in Silicon Valley and with people in Silicon Valley about what they knew during the time you worked for OnCore. Okay?

A. Okay.

Q. Okay. So -- and I understand you left OnCore in 2002; is that right?

A. I don't remember the exact date, but that sounds -- that sounds like the right range, yes.

Q. Or maybe earlier, but by 2002?

A. Yeah.

Q. Okay. So by 2002 did software developers in Silicon Valley know that large numbers of computers running different operating systems could be connected in a network?

MR. TONG: Objection. Foundation. Calls for speculation.

THE WITNESS: Yes, there were many examples of that scenario you just described by that timeframe.

BY MR. ANAPOL:

Q. And by 2002 did software developers in Silicon Valley know that servers running different operating systems could work together to provide a service?

MR. TONG: Objection. Foundation. Calls for speculation.

THE WITNESS: Yes, there are also examples of that by that date.

BY MR. ANAPOL:

Q. Can you give an example?

A. We discussed things with Chase Bank in New York, and they were using Solaris and Windows together to form various applications, Windows being the front end, Solaris being the data management in the back end, together forming a service.

Q. That reminds me. One follow-up question about OnCore: So was OnCore selling the operating system we discussed earlier by 2021?

A. Yes.

Q. By 2002 did software developers in Silicon Valley know that software could run inside of containers?

MR. TONG: Objection. Foundation. Calls for speculation.

THE WITNESS: Yes. The use of the term "containers" is broad, but if you allow yourself a broad definition of "containers," the answer is yes.

BY MR. ANAPOL:

Q. Did software developers in Silicon Valley by 2002 know that servers could host more than one container?

THE WITNESS: Yes.

MR. TONG: Same objection, belated.

BY MR. ANAPOL:

Q. By 2002 did software developers in

Silicon Valley know that putting an application in a container could prevent the application from accessing files in another container?

MR. TONG: Same objection. Calls for a legal conclusion.

I'm also objecting to this whole line of questioning again as getting into fact discovery that Mr. Donn Rochette is not obligated to answer during venue discovery.

BY MR. ANAPOL:

Q. Do you need me to repeat the question, Mr. Rochette?

A. Yes, please.

Q. Sure.

By 2002 did software developers in Silicon Valley know that putting an application in a container could prevent the application from accessing files in another container?

MR. TONG: Same objections.

THE WITNESS: Yes. In the context of Solaris zones, that is a true statement.

BY MR. ANAPOL:

Q. By 2002 did software developers in Silicon Valley know that putting an application in a container could prevent the application from interfering with applications in another container?

MR. TONG: Same objections. Calls for speculation. Foundation. Calls for a legal conclusion. Not venue discovery.

THE WITNESS: Again, in the context of Solaris zones, that would be an accurate statement. At least in the context of Solaris zones.

BY MR. ANAPOL:

Q. What about in the context of OnCore operating system?

A. Yes.

MR. TONG: Same objections.

THE WITNESS: Well, no, not in OnCore. In Trigence. Excuse me.

BY MR. ANAPOL:

Q. So did the OnCore containers not

confine the applications in the container?

MR. TONG: Objection. Calls for legal conclusions.

THE WITNESS: Yeah, it's difficult to answer.

The Unix applications running with Unix were not separated and could affect each other. Embedded applications were separated.

BY MR. ANAPOL:

Q. So, in other words, you could have multiple applications in one container?

MR. TONG: Objection. Vague.

THE WITNESS: In a Unix context, yes.

BY MR. ANAPOL:

Q. And the applications running in the Unix context on top of OnCore would be prevented from interfering with the applications running on the real-time portion of the operating system; correct?

A. Yes.

MR. TONG: Objection. Calls for a legal conclusion.

BY MR. ANAPOL:

Q. Sorry. Can you just repeat the answer, Mr. Rochette.

A. The answer is yes.

Q. And is it also true that putting an application in the Unix container in the OnCore operating system would prevent it from interfering with files outside of that container?

MR. TONG: Objection. Calls for speculation. Calls for a legal conclusion.

THE WITNESS: That is a true statement -- an accurate statement.

BY MR. ANAPOL:

Q. By 2002 did software developers in Silicon Valley know that containers could have their own root file systems?

MR. TONG: Objection. Calls for speculation. Leading.

THE WITNESS: That is an accurate statement in the context of zones and Trigence AE [garbled].

(The reporter requested clarification.)
THE WITNESS: Sorry.
That is an accurate statement in the context of Solaris zones, Trigence AE at least.
BY MR. ANAPOL:
Q. I want to clarify that because I'm asking about 2002.
So does that change your answer?
A. Oh.
MR. TONG: Objection. Asked and answered. Same objection as well.
THE WITNESS: In answer to your question, yes. I shouldn't use examples, but, yes, people knew that that was possible.
BY MR. ANAPOL:
Q. And are you familiar with a Unix command called chroot.
A. Yes.
Q. And did that Unix command allow a user to give an application its own root file system?
A. Yes.
Q. And do you know when chroot was

introduced?

A. It was introduced in the early Unix days before Solaris. It was introduced by the Bell Labs people when they first did Unix. It goes way back.

Q. So by the 1980s at least?

A. At least. Before then, yes.

Q. And are you familiar with something called BSD jails?

A. Yes. I forgot about those. Yes, I am.

Q. And what was BSD jails?

A. It was an open source response to the capabilities in Solaris zones. It was a way of obtaining very similar behavior exemplified by zones in an open source Unix capability that didn't require licensing yet. They're not exactly the same, but it was close enough.

Q. So is BSD jails another container capability?

A. Yes.

MR. TONG: Objection. Calls for legal conclusion to the previous question.

BY MR. ANAPOL:

Q. And do you know one way or the other if "BSD" stands for "Berkeley Software Distribution"?

A. It is my understanding that that is what it stands for, yes.

Q. And that's referring to the University of California, Berkeley, in Northern California?

A. That is my understanding, yes.

Q. By 2002 did software developers in Silicon Valley know that containers could run on top of a kernel that was residing outside of the caper?

MR. TONG: Objection. Calls for speculation. Foundation. Leading.

THE WITNESS: Yes.

MR. TONG: Calls for a legal conclusion. Sorry.

THE WITNESS: People understood that.

BY MR. ANAPOL:

Q. We're almost done with this line of questions, just so you know, Mr. Rochette. I

know it's repetitive in some sense.

By 2002 did software developers in Silicon Valley know that servers could limit the resources used by a container?

MR. TONG: Objection. Calls for speculation. Foundation. Leading.

THE WITNESS: Yes.

BY MR. ANAPOL:

Q. And did the OnCore operating system have the ability to limit resources used by a container?

MR. TONG: Same objections.

THE WITNESS: That was not the intent of the OnCore system, no. There was no explicit limiting of resources or -- in that -- in that architecture.

BY MR. ANAPOL:

Q. But the OnCore system would allow a real-time program to preempt an application running in a Unix container; correct?

A. That is correct. But not for -- it would not allow -- it would not enforce the

amount of memory, for example, that a Unix application would use. Nothing of that nature was done.

Yes, the really important thing was that an application running in a real-time context could preempt anything running in Unix and be able to respond immediately.

Q. And when the real-time application preempts the application in the Unix container, the real-time container is taking processing resources away from the container at that time?

A. Absolutely correct, yes.

MR. TONG: Same objections.

THE WITNESS: Sorry. I should wait for you to object.

Yes.

BY MR. ANAPOL:

Q. By 2002 did software developers in Silicon Valley know that servers could monitor and log a container's resource usage?

MR. TONG: Same objections.

THE WITNESS: Yes.

BY MR. ANAPOL:

Q. And by 2002 did software developers in Silicon Valley know that containers could have their own IP address separate from a host?

MR. TONG: Same objections.

THE WITNESS: I'm not 100 percent sure of that. That was very difficult -- that's very difficult to accomplish, the IP address, and I don't -- honestly don't recall if that was considered sort of commonplace.

BY MR. ANAPOL:

Q. But would it surprise you if someone had that capability before 2002?

MR. TONG: Objection --

THE WITNESS: It would not surprise me --

MR. TONG: -- foundation.

THE WITNESS: -- no.

Oh, sorry.

BY MR. ANAPOL:

Q. Can you repeat the answer, Mr. Rochette.

A. It would not surprise me, no.
Q. And by 2002 did software developers in Silicon Valley know how to move software functionality into a shared library?
MR. TONG: Objection. Calls for speculation. Foundation. Legal conclusion. Not venue discovery. Leading.
THE WITNESS: Yes.
BY MR. ANAPOL:
Q. And do you know whether Trigence released its container software before or after Solaris zones?
A. I don't recall whether anything in Windows or Linux was released before or after zones, no. I don't recall the dates, no.
Q. Have you ever heard of Docker?
A. Yes.
Q. And what is your understanding of Docker?
A. Docker is a container technology that is in wide use today.
Q. And when did you first hear about

Docker?

A. Maybe after the Trigence days, at the end of Trigence -- my time at Trigence. I don't recall a date.

Q. Have you personally used Docker?

A. Yes.

Q. In what context?

A. Creating containers. Allowing -- for development purposes. For production purposes. For monitoring purposes. We use Docker quite a bit.

Q. When you say, "We have used Docker," who are you referring to?

A. Oh, I'm sorry. Multiple development teams that I've been a part of. We have all used Docker.

Q. Which companies that you've worked at have used Docker?

A. AppFirst. ScienceLogic. I worked as a contractor for the US Air Force for a while. And Cribl most recently.

Q. Okay. So all four of those

organizations used Docker?

A. Correct.

Q. And, Mr. Rochette, have you ever heard of containerd?

A. Yes.

Q. What's your understanding of containerd?

A. Containerd is a process that runs in the background as a part of Docker and is -- implements a large portion of the Docker functionality.

Q. And do you have an understanding of where Docker originated?

MR. TONG: Objection. Calls for speculation.

THE WITNESS: I don't, actually. I know it's open source, but I don't know where or what group originated it, no.

BY MR. ANAPOL:

Q. You mentioned Cribl in one of your earlier answers.

Where is Cribl located?

A. Cribl is located in San Francisco.
Q. And do you know if Cribl is using Docker in San Francisco?
A. Yes.
Q. And did you visit San Francisco when you worked for Cribl?
A. Yes.
Q. How often?
A. Not very often. Once a year.
Q. And are there a lot of people in Northern California who know about Docker?
A. Yes.
MR. TONG: Objection. Calls for speculation.
BY MR. ANAPOL:
Q. And Cribl is the most recent company that you worked at?
A. That is correct.
Q. Do you still work there?
A. No. I am retired.
Q. Oh, right. You mentioned that earlier.
When did you retire from Cribl?

A. 1st of September two years ago.
Q. 2022?
A. Yes.
Q. And how does Docker compare to the technology developed at Trigence?
MR. TONG: Objection. Calls for legal conclusions. And Daubert.
THE WITNESS: It is very similar. Docker containers are quite similar to what is described in the Trigence patent. There are -- there are a lot of differences. You know, any software developer can describe those differences to you. But overall the ability to create multiple containers with, you know, multiple disparate applications running in them is consistent.
BY MR. ANAPOL:
Q. And what do you mean by "disparate applications"?
A. Primarily referring to applications that would require a specific version of an operating system. So something that would run

on -- something that was originally running in a Windows 10 environment that would be placed in a container and could run in a Windows 12 environment. That's what I mean by "disparate," meaning --

Q. So the environments are disparate, meaning different versions of the same operating system?

A. Different versions of the same operating system, correct.

Q. Okay. So did you make the decision to use Docker at the four companies you mentioned, or did somebody else make that decision?

A. Oh, no, somebody else made the decision. It was -- yeah.

Q. Did you ever suggest to any of those four companies that they use Trigence AE instead of Docker?

A. No.

Q. Why not?

A. Because -- the primary reason is because of the difference in the way the

Trigence AE is implemented, which requires function overlays and shared libraries and so forth. It's difficult for larger companies' environments to adopt that kind of approach.

Q. Why?

A. Because it modifies the way -- there are modifications in the way in which an application would run that are reliant on private, non-open-sourced libraries and so forth that make it very difficult for someone to be able to validate -- and you need support and things like that.

So it's much more difficult to adopt that kind of an approach. And I think one of the reasons -- it's one of the reasons why Docker is very typically chosen over an approach like Trigence AE.

Q. And did you ever think that the companies you were working for that use Docker needed a license to the '814 or '058 patents?

A. No. Never crossed my mind. It never crossed my mind that they were similar.

Q. And do you have an understanding of whether any of the four companies you worked at that use Docker were also using containerd?

A. If you were using Docker, you were using containerd.

Q. So all four of those companies were also using containerd?

A. Yes.

Q. Okay. I want to talk a little bit more about the disparate computing environments that you mentioned.

A. Okay.

MR. ANAPOL: And, Billy, could you please pull up Exhibit 1010 for us, and scroll down to Column 2, please. And around lines 17 to 19, can you blow that up for us.

Q. So in the '814 patent, Mr. Rochette, do you see that there's this definition of "disparate computing environments"?

A. Yes.

Q. Do you know what this means?

A. Yeah. It's -- it's worded like a

lawyer would word it. Sorry.

It is -- it's meant to speak to the ability for applications that are unrelated to each other to run in a common compute platform.

Q. So it says, "Environments where computers are stand-alone or where there are plural computers and where they are unrelated."

So does "unrelated" refer to the computers or to the applications running on them?

A. In this statement it refers --

MR. TONG: Objection. Calls for a legal conclusion.

THE WITNESS: Oh, sorry.

BY MR. ANAPOL:

Q. Go ahead, Mr. Rochette?

A. In this statement it refers to the computer platform -- the computer platforms.

Q. And what does it mean for computer platforms to be unrelated?

MR. TONG: Objection. Calls for a legal conclusion.

THE WITNESS: It would be having a -- there would be stand-alone, having -- presumably having a network connection where information could be exchanged, but they are not necessarily related to each other.

If you go back to the example of a corporate business application where it's using Windows as a front end and some Unix or Solaris background for data would be an example of that. They are unrelated to each other, Windows is Windows, Unix is Unix, and they are stand-alone, but -- and [unintelligible] of things unrelated that can be made to be related by applications that talk to each other.

Does that help? I'm not sure I'm answering the question.

BY MR. ANAPOL:

Q. Yeah, I'm just trying to understand how you tell whether two computers are stand-alone or unrelated.

Do they have to be running different operating systems?

A. No. They would just be separate instances.

If you look at a rack of computers, each hardware platform is a separate stand-alone device. Even if they're already all running the same hardware and all running the same operating system, they are different from each other.

Q. Okay. So they're considered unrelated even if they're all in the same place, they're all connected to the same network and they're all running the same operating system?

A. Yes.

MR. TONG: Objection. Calls for a legal conclusion.

THE WITNESS: Oh, sorry.

One way to differentiate that is that they would all have a different network address would make them standalone and separate.

BY MR. ANAPOL:

Q. Okay. You reside in Iowa; correct?

A. Yes.

Q. And you're retired now; right?

A. Yes.
Q. So you're not employed in any state outside of Iowa; is that right?
A. That is right.
Q. And are there any states outside of Iowa that you regularly travel to for business?
A. Not for business. Only for personal.
Q. And before you retired, you worked for Cribl; right?
A. That is correct.
Q. Which was based in San Francisco?
A. Yes.
Q. And sorry if you already answered this. I can't recall.
How often did you visit San Francisco when you worked for Cribl?
A. About once a year. Not frequently.
The company was remote. So we had people in Europe and all across the United States, and we all worked remotely.
Q. Okay. I just want to get back to Docker --

A. Okay.
Q. -- and then hopefully we'll be able to wrap up here.
So Docker, based on your experience -- I'm sorry. I didn't mean Docker. I meant Solaris zones.
A. Okay.
Q. So Solaris zones, based on your experience, can operate in disparate computing environments?
A. Yes.
MR. TONG: Objection. Calls for a legal conclusion. Daubert.
BY MR. ANAPOL:
Q. And Solaris zones was capable on running of on a plurality of servers, meaning multiple servers?
A. Yes.
Q. And those servers would include a processor; right?
A. Yes.
Q. And an operating system; right?

A. Yes.
Q. And a kernel; right?
A. Yes.
Q. And local system files compatible with the processor?
A. Yes.
Q. And the server could provide executable applications related to a service?
MR. TONG: Objection. Calls for legal conclusions. Daubert.
THE WITNESS: Yes.
BY MR. ANAPOL:
Q. And Solaris zones allowed applications to be executed in a secure environment?
A. Correct.
MR. TONG: Same objections. And leading.
THE WITNESS: Yes.
BY MR. ANAPOL:
Q. I will rephrase the last question.
Could a server running Solaris zones provide executable applications relating to a

service?

MR. TONG: Same objections.

THE WITNESS: Yes.

BY MR. ANAPOL:

Q. And could a server running Solaris zones execute applications in a secure environment?

MR. TONG: Same objections.

THE WITNESS: Yes.

BY MR. ANAPOL:

Q. And could a server running Solaris zones store multiple secure containers in memory?

MR. TONG: Same objections.

THE WITNESS: Yes.

BY MR. ANAPOL:

Q. And could those multiple containers have applications and system files in them?

MR. TONG: Same objections.

THE WITNESS: Yes.

BY MR. ANAPOL:

Q. And would those containers use the

kernel from the operating system hosted on the server?

MR. TONG: Same objections.

THE WITNESS: Yes.

BY MR. ANAPOL:

Q. And that kernel would be on the server regardless of whether it was running any containers; right?

MR. TONG: Same objections.

THE WITNESS: Yes.

BY MR. ANAPOL:

Q. Would the kernel be running on the operating system regardless of whether the server was running any containers?

MR. TONG: Same objections.

THE WITNESS: Yes. The kernel would run first and foremost, yes, in all instances.

BY MR. ANAPOL:

Q. And would the system files in the containers be compatible with the kernel?

MR. TONG: Same objections.

THE WITNESS: That is a major

difference. In zones they would have to be compatible with the kernel and the version of the kernel being used, yes.

BY MR. ANAPOL:

Q. And did each container have its own kernel or did they share a kernel?

A. No --

MR. TONG: Same objections.

THE WITNESS: -- one kernel shared by -- oh, I'm sorry.

One kernel shared by all applications.

BY MR. ANAPOL:

Q. And could a container contain system files that were also present outside of the container?

MR. TONG: Same objections.

THE WITNESS: I'm sorry. Could you clarify that.

BY MR. ANAPOL:

Q. Yeah. So imagine you have a service running outside of the container, like, you know, you have an instance of Apache, for

example.

A. Okay.

Q. And then you also have an instance of Apache running inside the container.

A. Uh-huh.

Q. Could you have system files, like, the configuration file for Apache, one inside the container and one outside the container?

A. Yes.

MR. TONG: Objection. Incomplete hypothetical. Leading. Daubert. Calls for a legal conclusion. Foundation.

THE WITNESS: Yes. That's actually a common scenario.

BY MR. ANAPOL:

Q. And --

A. A configuration file is a very common scenario.

Q. And would the instance of Apache inside the container use the configuration file inside of the container instead of the one outside of the container?

MR. TONG: Same objections.

THE WITNESS: Yes.

BY MR. ANAPOL:

Q. And would the applications inside of a container on Solaris zones be able to access software in another container?

MR. TONG: Same objections.

THE WITNESS: No.

BY MR. ANAPOL:

Q. And would each of those containers be able to have its own root file system?

MR. TONG: Same objections.

THE WITNESS: Yes.

BY MR. ANAPOL:

Q. And do you know if Solaris zones allowed resource limits to be placed on a container?

A. Yes, it is.

MR. TONG: Same objection.

BY MR. ANAPOL:

Q. Mr. Rochette?

A. Yes.

Q. And would applications inside of a container be prevented from accessing system files outside of the container?

MR. TONG: Same objections.

THE WITNESS: Yes, that is correct.

BY MR. ANAPOL:

Q. And do you know how a container would be created in Solaris zones, in other words, how you would get the necessary files into the container?

A. There were very specific command line options that you used with Solaris zones. A zone_create command and a zone command for specifying resources, a zone command for specifying files that were used in the zone, and things like that. So you would -- you would configure these things using a command line.

Q. So did Solaris zones have the ability to create a container -- basically take an application that's already running outside of a container and then make a copy of it inside of a container?

MR. TONG: Same objections.

THE WITNESS: You would make a copy -- oh, sorry. Excuse me. Sorry.

Yes, you would make a copy of it. You would create the zone, then you would copy files as needed, very likely you'd have to modify the configuration of something, and then you could start your application.

BY MR. ANAPOL:

Q. And when you would copy files into the container, it would be applications and system files, or no?

A. Application files for sure. System files would be optional, depending on what you were doing. You might need some system files, you might not, depending on what the application was and what you were trying to accomplish.

Q. And when you made that copy, you would still have the original version outside of the container; right?

A. You would -- it would be possible, yes.

Q. I think you mentioned earlier that

Solaris zones had the capability of moving certain functionality that would normally be in an operating system kernel into a shared library; is that right?

MR. TONG: Objection. Mischaracterizes. Leading.

THE WITNESS: Yes, there were some capabilities that were represented outside of the kernel in user mode, correct.

BY MR. ANAPOL:

Q. What capabilities were those?

A. File system separation. The use of something like the chroot command, or the chroot command you referred to earlier. The -- oh, I can't think of other examples. I mean, there were several examples of it.

Q. And how did you learn that Solaris zones had that capability?

A. Sitting down in front of a Solaris operating system and using it. Reading the documentations and using the capabilities and testing it.

Q. And how did you determine that those capabilities were implemented in shared libraries rather than the kernel?

A. If you go to an application and -- you would execute the command called LDD. LDD shows you what libraries are required by a certain executable, and it would -- it's very clear -- clearly shown what libraries are referenced, and you could then look at the libraries and very clearly see what is implemented in those libraries.

Q. So do you know if there was any shared library functionality in Solaris that was also in the kernel, in other words, the shared library functionality was a duplicate or performing a similar function to what was in the kernel?

MR. TONG: Objection. Foundation. Leading. Calls for a legal conclusion.

THE WITNESS: There were some networking capabilities and shared libraries and some -- for example, that would define what IP

address is used, define what routing is used for a container separate from the kernel and was -- would be accessed by applications before going to the kernel.

So you would have had very similar capability in the kernel that was accessed by applications in these zones before going to the kernel.

BY MR. ANAPOL:

Q. So just for the sake of a concrete discussion, can you think of an example of a shared library that we could talk about that had this operating system functionality?

A. I can't recall the names of them in Solaris. They would have been a libnetwork.so or lib -- something with the name "network" in it or "net." And it would have the network -- the things that would otherwise -- the things that were also present in a kernel, like defining an IP address, like defining routes, like defining -- not TCP protocol, but these would have been used by zones to configure the

network capabilities present in that zone or container for use by the application.

I don't remember exactly the names. It's a long time ago, but there were shared libraries used by Solaris that did those things.

Q. So for the networking shared library, if there were two applications using that shared library, would they each have their own instance in the sense of, you know, internal data used by the shared library having two different copies of that data?

MR. TONG: Objection. Foundation. Leading. Calls for a legal conclusion. Daubert.

THE WITNESS: The way it works is the code for that library in question, once loaded into memory, is shared by all applications. That's the -- hence, the term "shared library." And -- so multiple zones could make use of the same networking library.

The difference is that the data that they use is distinct. So the actual code that

gets executed is common across zones; the data that they use is separate for each zone.

And any writes that would happen into memory -- oh, gosh, at the risk of going into detail would be -- would implement an operating feature called copy-on-write, which would create a different instance of a page in memory, but it's specific to that zone.

Sorry. I think I went too deep.

BY MR. ANAPOL:

Q. And did the Trigence AE approach to shared libraries work the same way?

A. Yes.

Q. So are you aware of any differences between how your '058 patent uses shared libraries versus how the Solaris zones used shared libraries?

A. There are fundamental differences, yes.

Q. Okay. What are the differences?

A. I'm pausing to think about how to make it very simple.

The most -- the biggest difference or

the most salient difference is the use of the library preload. I referred to it earlier, LD_Preload and AppInit_DLL. The Trigence patent defines the use of these capabilities by and from an operating system so that a Trigence-specific library gets loaded independent of the application or in addition to the libraries required by the application. And these function overlays I referred to earlier are used a lot. It's the foundation to the way that technology is defined.

This is not used in zones. Zones did not do any of that. Where they required, for example, different -- where they required, for example, specific behavior for one zone versus another zone, they were able to rely on kernel internals in Solaris do that. Trigence AE is not able to do that and therefore would rely on preload and function overlays in order to accomplish those kinds of things.

Q. Okay.

A. Really big difference. There are

others, but that's the biggest.

Q. And do you know if the claims of the '058 patent reflect that difference?

MR. TONG: Objection. Daubert. Calls for a legal conclusion.

THE WITNESS: When I reviewed the patents in preparation for this discussion, I did not look at the claims. I don't -- I couldn't answer that effectively right now.

MR. ANAPOL: All right. So I'm almost done. I probably have a couple more questions, but I want to give you an opportunity to take a lunch break if you want one.

THE WITNESS: If it's okay with everybody else, I'd rather just finish.

MR. ANAPOL: Okay. All right.

THE WITNESS: If everybody else is okay.

MR. ANAPOL: We've been going about an hour. So, Audra, do you want to do five minutes or ten minutes for a break?

We'll do a ten-minute break. I'll have

a few questions to wrap up. Mr. Tong might have some questions after that, but hopefully we can just wrap this up before you need to eat lunch.

THE WITNESS: Thank you.

THE VIDEOGRAPHER: Okay. Going off the record. The time is 12:59.

(Recess taken.)

THE VIDEOGRAPHER: We are back on the record. The time is 1:10.

BY MR. ANAPOL:

Q. Okay. So before the break, we were talking about Solaris zones.

Do you remember that?

A. Yes.

Q. And so, as you sit here today, are you able to identify any difference between the claims of the '058 patent and what was in Solaris zones?

MR. TONG: Objection. Calls for a legal conclusion. Daubert.

THE WITNESS: I would have a difficult time being specific about claims because, as I

said, I didn't review the claims. I can -- I am comfortable with differences in technology or differences in implementation, but the specific claims I would have to -- I couldn't answer right now.

BY MR. ANAPOL:

Q. So when we go back to the court, just so you understand, there's going to be some briefing where we explain, you know, our position and the other side's position, and when we are describing to the court what happened here, I want to be able to tell the court whether you're what we call a willing witness or an unwilling witness.

So an unwilling witness would be someone who is only going to come if they're subject to subpoena that requires them to come to the trial, and a willing witness is someone who would come willingly even if they're not compelled to come.

And so would you put yourself in the willing or unwilling category?

A. In all honesty, I would put myself in the unwilling category.

Q. Okay. And then I just wanted to back briefly to the Docker issue.

A. Okay.

Q. So you used Docker and containerd at four different companies; right?

A. Correct.

Q. And you are the named inventor on both the '814 patent and the '058 patent; correct?

A. Yes.

Q. The first named inventor.

A. Yes.

Q. And it never occurred to you at any of those four companies that your company needed a license from Trigence or VirtaMove to use Docker or containerd?

MR. TONG: Objection --

THE WITNESS: It never occurred to me.

MR. TONG: -- leading.

THE WITNESS: It never occurred to me, no.

BY MR. ANAPOL:

Q. Do you think now that Cribl needs a license to the '814 or '058 patent to use Docker and containerd.

MR. TONG: Objection --

THE WITNESS: Absolutely not.

MR. TONG: -- leading. Calls for a legal conclusion.

THE WITNESS: I'm sorry. I keep doing that.

Absolutely not, no.

MR. ANAPOL: No further questions.

EXAMINATION

BY MR. TONG:

Q. Mr. Rochette, it'll be my turn to ask a few more questions.

A. Okay.

Q. First of all, I want to thank you for your time here today.

In preparation for this deposition, have you been in communication with Amazon or

its counsel?

A. I spoke with Mr. Anapol very briefly to clarify whether or not -- to clarify what I needed to be prepared -- what I needed to bring in; if there was any response that was required that was -- yep.

So I had one conversation on the phone just to make sure I was doing what I was supposed to be doing.

Q. Did you speak with anyone else other than Mr. Jeremy Anapol?

A. No.

Q. About how long did you speak with Mr. Anapol?

A. Oh, I think the conversation might have lasted ten, fifteen minutes is all.

Q. Did you speak with any other lawyers from other firms?

A. No.

Q. Okay. And did you try and retain your own attorney to defend you in this deposition?

A. No.

Q. Did Amazon offer you any type of compensation or paid role as a consultant?
A. No.
Q. Did you have any email exchanges either with Amazon or its attorneys or other representatives?
A. The only email I've had is a clarification of which Zoom I could use.
Q. Did -- on your -- when you had that discussion with Mr. Anapol, what topics did he tell you to be prepared about?
A. He explained that there was a -- I think he described it as a motion before the court about change of venue, and he explained why they were doing that.

He explained that this was in reference to the two patents, making sure that I was comfortable with what those patents were.

And, yeah, that's about it.
Q. Did he discuss any lines of questioning with you?
A. No.

Q. Did he talk about how you should answer any of the questions that he was going to ask today?

A. No. He was very careful to say just be honest and answer to the best of your ability. That was my recollection.

Q. Did you have any communications with Amazon today during the breaks in this deposition, either Amazon or its representatives, like Mr. Jeremy Anapol?

A. No.

Q. Did you read the patents in preparation for this deposition?

A. Yes. Like I said, I did not read over the claims, but I did read over the patents, yes.

Q. If you didn't want to be involved with this lawsuit, why did you prepare for this deposition and agree to, you know, so willingly join us today?

MR. ANAPOL: Objection. Argumentative. And mischaracterizes testimony.

And as you know, Peter, he was subject to subpoena, so he was required by law to attend.

MR. TONG: Okay. No speaking objections, please.

MR. ANAPOL: You can answer, Mr. Rochette.

THE WITNESS: I was required by law to attend based on federal subpoena, yeah.

BY MR. TONG:

Q. So you recall what you said was you speaking with Mr. Nigel Stokes?

A. Yeah.

Q. That was through LinkedIn; right?

A. Correct.

Q. Okay. It wasn't actual speaking, like, by telephone, was it?

A. Yes, we had a phone conversation. He contacted me over LinkedIn, he asked for permission to call me, and we had a -- we had a telephone conversation, a brief telephone conversation.

Q. Okay. And during that conversation, you generally expressed that you did not want to be involved with this case; correct?

A. That is correct.

MR. ANAPOL: Objection. Leading.

BY MR. TONG:

Q. And so today you heard me object several times that this deposition is supposed to be limited to venue facts.

Do you remember that?

A. I do recall you saying that, yes.

Q. Why did you keep answering questions after that objection?

A. Because I was told -- informed by other counsel that I was required to answer it.

Q. And was that Mr. Anapol --

A. Yes.

Q. -- who informed you of that?

A. Yes.

Q. Did he tell you that you had to cooperate with him other than by showing up at this deposition today?

A. I don't understand that question.
Q. Did he tell you that the subpoena required you to cooperate with him in any other way outside of this deposition?
A. No.
Q. Does Amazon represent you in any capacity?
A. No.
Q. Does Mr. Jeremy Anapol represent you -- or any other member of his law firm, Knobbe Martens, represent you in any capacity?
A. No.
Q. If you recall, earlier today we looked at the '814 patent, and --
A. Yes.
Q. -- you said it was generally related to containers.
A. Yes.
Q. At a high level, what was it about the use of containers that you thought was inventive and worth filing a patent on?
A. The ability to create separate

application environments or environments for applications to run in Windows and Linux.

Q. And you thought that was inventive at the time you came up with it; right?

A. Yes.

Q. Did OnCore do that already by the time that you had filed the application for the '814 patent?

A. Not -- they're not the same. OnCore created -- they used the term "container." You saw what was displayed earlier. They used the term "virtual containers," and they are; they're virtual containers. But they're very different.

They are -- the intent of what OnCore did was to create priority preemptable, embedded applications running side by side with Linux. What this patent describes is very different from that.

Q. Could you elaborate a little bit more on what this patent describes that is very different from the virtual containers of OnCore.

A. The '814 patent describes an ability to

utilize library preloads, function overlays in order to support a environment in which applications with run in secure containers, in separate containers that the OnCore technology did not use library preload or function overlays at all.

Q. Thanks.

By the time you had filed the '814 patent, do you think the technology was -- of the '814 patent -- was that different from what Solaris zones was doing at the time?

A. Yes, it was different. And the way in which it was implemented was different, and the -- much of the features were different. But, yeah, there were differences.

The primary difference is the technology defined in the '814 patent could be used outside of Solaris. Zones is specific to Solaris.

Q. Other than that the technology of Solaris zones was specific to Solaris, were there any other ways that zones and the

technology of the '814 patent were different?

A. Well, the fundamental design was vastly different. The architecture was vastly different.

I feel like I'm repeating myself, but zones relied on specific Unix commands and modifications to the kernel in Solaris in order to implement those containers. What's defined in the '814 patent does not rely on changes to the kernel and does not rely on specific Unix commands to implement these -- the containers that are described.

Q. And that would have been -- modifying the kernel in something like Solaris, that would be very difficult to do for an average programmer if that programmer did not have access to the Solaris kernel source code; correct?

MR. ANAPOL: Leading.

THE WITNESS: It would be very difficult. Virtually impossible.

BY MR. TONG:

Q. My previous question asked you about the difference between the '814 patent and the virtual containers of OnCore.

Did those differences exist at the time that you had filed the provisional applications that the '814 patent claims priority to?

A. I'm sorry. Make sure I understand your question.

Did the differences between what the OnCore technology defined and what's defined in the '814 patent exist at the time the provisional was submitted?

Yes, those differences did exist at that timing.

Q. And similar question for Solaris zones: Did the differences between Solaris zones and the technology of the '814 patent exist at the time that you filed the provisional applications that led to the '814 patent?

A. As I stated earlier, I don't recall the timing when zones were announced and released

relative to the provisional patent application. I don't recall the timing of that. It was close.

Q. So earlier you had discussed a meeting with the folks at Solaris zones.

Do you remember discussing that meeting?

A. Yes.

Q. And initially they thought that you or someone else had gained access to the zone source code?

A. That was their impression when we went into the meeting, correct.

Q. Okay. And then you explained to them that you had, in fact -- well, what did you explain to them in response to that accusation?

A. We explained in a lot detail how Trigence AE worked, why it worked, what we did, how to make it work, and they were able to see in that discussion and whiteboarding that what we were doing was able to be accomplished without access to Solaris internals.

Q. Initially were they skeptical when you denied accessing Solaris source code?

A. I suppose they were.

Q. And you had to more or less convince them and dissolve that skepticism through lots of details and discussions of how Trigence AE worked; correct?

A. Yes.

MR. ANAPOL: Objection. Leading.

BY MR. TONG:

Q. So when you were talking with Mr. Anapol, you had said that at your previous job at OnCore you only very loosely worked on containers.

Do you remember using the words "very loosely" in an answer?

A. Yes.

Q. What did you mean by you only very loosely worked on containers?

A. The use of the term "containers" has taken on today -- in today's understanding of the technology has taken on a fairly specific

and well-understood understanding of what that means.

When the OnCore work was done, there was not such an understanding of what "containers" mean. It was a new kind of topic, and there was not a common understanding of it, and so we could -- the website and marketing people and salespeople could use the term "containers," and it was acceptable.

And so that's what I mean by a loose -- use of the word "containers," you would not be able to do that. It would -- you would not be able to use that term today for what OnCore is doing or had -- did do. That's what I was trying to imply.

Q. Okay. How was OnCore using the term "containers" back then?

A. OnCore would take a microkernel, would deploy embedded real-time applications using that microkernel and, in parallel with that, would employ Unix or Linux in an application. So a real-time application would be running as a

peer to all of Unix.

And so the term "container" would be used to describe the real-time application running as well as Unix as an application, and all of the applications on top of Unix would be -- that's how they -- the term "container" -- "virtual container" was used in the OnCore context.

Q. When you were at OnCore, did you ever use the word "docking" or "docker"?

A. I don't recall the use of that term in the OnCore context.

Q. Before you filed either of the '058 or '814 patents at issue in this case, did you hear anyone else use the word "docking" or "docker"?

MR. ANAPOL: Objection. Vague.

THE WITNESS: I don't recall whether or not "docker" was a commonly used -- at the time that was used. I don't recall.

MR. TONG: Let's turn to the '058 patent for a little bit.

Could we please put Mr. Rochette back

in the main window?

THE VIDEOGRAPHER: Do you not want me to share?

MR. TONG: No, it's not necessary.

Q. So earlier do you remember discussing how this patent related to shared libraries?

A. Yes.

Q. Now, you're not claiming to be the inventor of shared libraries; correct?

A. That is correct.

Q. The '058 patent covers and discloses more than just the use of a shared library by itself; correct?

A. That is correct.

Q. Okay. So what did you think was inventive about the use of shared libraries in the '058 patent?

A. A number of things. The biggest thing was the ability to implement things that were at the time only implemented in the kernel. So you could, for example, provide an application that required an older version of Windows. Say

something ran on Windows 8; you could run it on Windows 10.

The -- it would be required that that application get specific files and libraries used only by Windows 8 that not are not available on Windows 10. So you had to ensure, when a file was opened, that you were able to use these shared library techniques in order to open the file related to that application -- required by that application, not the file provided by the operating system.

Another example of that would be an ability to have an application use an IP address specific to that application or container which is separate from other IP addresses. So you had to again use code in these shared libraries in order to basically ensure that the application thought it had one IP address.

And that's where that whole use of the term "critical system elements" in the '058 patent comes into play. And you were taking functionality that would otherwise only

be available in a kernel and putting that into libraries that you caused that application to use.

Q. So you said that that was the biggest thing.

Were there any other inventive concepts related to the '058 patent that you felt like you had invented?

A. Well, the ability to do --

MR. ANAPOL: Objection. Calls for -- sorry. Sorry, Mr. Rochette.

Calls for a legal conclusion.

Go ahead.

THE WITNESS: The ability to do any of these things we've been discussing in a Windows context at the time was very inventive. There was no way to do any of this in Windows without what was described at the time in these technologies.

BY MR. TONG:

Q. Let's talk for a minute about possibly conflicting libraries.

If I had an application running in an old operating system that needed to use a particular library file called, let's say, ABC, and I moved that application to a newer operating system: What was the solution to allow the moved application to still use the library file ABC from the old operating system rather than having it look at ABC from the new operating system?

MR. ANAPOL: Okay. Mr. Rochette, before you start, I have a couple objections, so I don't want to interrupt you.

Calls for opinion. Outside the scope. Lacks foundation. Incomplete hypothetical.

Go ahead.

THE WITNESS: The way it works is you would use AppInit_DLL or LD_Preload to load a library that -- again, as I described earlier, that would have an effect on the namespace.

So when the application opened ABC library, the function overlay would cause the open to actually be implemented in one of these

critical system libraries or the Trigence library. The Trigence library would go and look for a file by that name in container-specific sources, and if it found it, it would use that. Even -- and if it didn't find it, then it would fall back to use the one in the operating system.

So if ABC -- if the library ABC existed in the container, it would use that instead of the operating system one, and that's how versions of the older operating system code and configuration data would be utilized by the application.

MR. TONG: Could we turn back, please, to page -- Exhibit 1012.

THE VIDEOGRAPHER: Do you want me to share this time?

MR. TONG: Yes, please do.

THE VIDEOGRAPHER: Okay. I'm sorry which page?

MR. TONG: Exhibit 1012. I think this was the --

THE VIDEOGRAPHER: Oh, 1012. My bad.

MR. TONG: Yeah.

Q. If I recall, we had looked at page 114 together.

Mr. Rochette, do you remember looking at this?

A. Yes.

Q. Okay. Do you think that what you had described in the '058 patent is different from what is being described in this paper titled "Shared Libraries in SunOS"?

MR. ANAPOL: Objection. Vague as to "what you had described in the '058 patent."

THE WITNESS: Yeah, it's a very broad question.

This is -- this paper is describing what your libraries are and the use of shared libraries and examples of how to use them. As you previously pointed out, the '058 patent is not claiming, you know, the invention of shared libraries.

It is using shared libraries and the

way in which shared libraries are utilized and the dynamic loader in order to accomplish these -- you know, in order to accomplish certain things, but it's not changing what shared libraries are or how they're loaded or anything like that. So it's making use of what your libraries are and how they are used.

I'm not -- am I answering the question? Is that -- I don't --

BY MR. TONG:

Q. You are. Thank you very much.

A. Okay.

Q. And what was the command to -- that you had mentioned earlier, I think you said to preload a library?

A. To preload a library, you used a function in the operating -- in Unix operating systems you used a command called LD_Preload. In Windows you used a feature called AppInit_DLL.

Q. And in the '058 patent you're not claiming to invent a preload command; correct?

A. That is correct. We're utilizing a preload command to accomplish what's described.

Q. Thank you.

By the time you had filed the provisional applications that led to the '058 patent, was OnCore doing -- or performing the technology that was described in the '058 patent?

MR. ANAPOL: Objection. Vague as to "the technology described in the '058 patent." Vague as to time. And lacks foundation.

THE WITNESS: Yeah, it's my understanding that OnCore was no longer a company. It was out of business by the time the provisionals were submitted for the '058 and '814. There was no OnCore. It was gone.

BY MR. TONG:

Q. At any time before the filing of the provisional patent that led to the '058 patent, did OnCore perform the technology that you had described as being disclosed in the '058 patent?

MR. ANAPOL: Foundation.

THE WITNESS: Again, in the broadest use of the term "containers," it can be conceived that they were using containers. But there was nothing -- did not specifically, no. The specifics defined in both the '814 and '085 patents are very different from what OnCore was doing, but in the broadest use of the term "containers," they both had containers.

BY MR. TONG:

Q. And similar question for Solaris zones: At any time before the filing of the provisional patent that led to the '058 patent, did Solaris zones perform the technology that you had described as being disclosed in the '058 patent?

A. Again, there's two parts to that answer, if I understand the question correctly.

No. 1, I don't recall the timing. As I've said, I don't -- I just don't recall when zones were announced and published relative to when the patents were -- when the provisionals were submitted. I just don't recall, No. 1.

No. 2, there are -- there's similar --

both zones and the '814 patent define "containers." There are very fundamental differences in how that's accomplished.

Q. Do you remember identifying two people that you had -- sorry. Strike that.

So you had mentioned talking to several people who worked for Solaris zones at a meeting.

Do you know where those people live today?

A. No. Not today, no.

Q. Not even generally in terms of a city or state?

A. No. I'm not aware.

Q. And I think you had mentioned talking to John Beck and David Comay, or at least that you recognized their names.

A. Yes.

Q. Do you know what city or state they live in today?

A. No, I do not.

Q. We had talked about interposition

earlier.

A. Yes.

Q. Do you remember that?

A. Yes.

Q. And you described interposition as "functionally the same as a function overlay."

What did you mean by "functionally the same"?

A. It's my personal understanding; right?

So, I mean, other people can have their own view of it, but they are accomplishing the same thing. There are a couple different ways in which it can be accomplished, but they are doing the same thing.

Q. So what's different?

A. Well, again, they're doing the same thing. You can -- you can cause that behavior to be accomplished in a couple different ways. The paper from Australia showed one way. LD_Preload is another example of doing that.

The end result is the same where -- when an application is running and it makes a

call to a function, you are causing that call to be resolved by -- in a library that you have caused to be loaded instead of the system library; right?

So in the example of Read, you load a library that implements Read, and you cause the loader to load that library either by linking directly to it or using LD_Preload when that -- there's also a function called Read in the system library called libc. And without the function overlay, without the interposition, when the application calls Read, it will call a function in libc.

However, with interposition/function overlay what you've done is you've caused that application to call a function called Read in the library that you've specified and not in libc.

Q. What was the primary use case for preload at the time of, say, 2002?

A. One --

MR. ANAPOL: Foundation --

THE WITNESS: Oh, sorry. Go ahead.

MR. ANAPOL: Foundation and vague.

THE WITNESS: There are multiple.

No. 1, you are using preload to cause a library to be loaded that wouldn't otherwise be loaded.

No. 2, you are causing -- you are enabling, by the act of loading that library, interposition/function overlay. You are then using function overlay to accomplish direction of files to the container, specific IP addresses and things of that nature that are described in the invention.

BY MR. TONG:

Q. To be clear, in 2002 are you aware of others who used function overlay in the context of containers and things of that nature as described in the invention, or was that -- or were you just -- sorry. Let me rephrase that.

My question was what was the primary use case for preload in the time of, say, 2002. The part of your answer, which was, you're then

using function overlay to accomplish direction of files to the container-specific IP addresses and things of that nature.

Was that part of your answer to something that was a primary use case for preload in 2002?

A. To the best of my understanding, there were examples of preload being used in 2002. You could -- you could find a number of examples.

To the best of my understanding, there were not examples of preload being used to redirect files and accomplish the networking things. It wasn't used in that nature or in that scope, to the best of my recollection.

Q. Okay.

A. You see the difference I'm trying to say?

LD_Preload -- preload was being used, and there are -- there were examples of it, but not in the context of creating a container-type environment --

Q. Got it.
A. -- is my recollection.
Q. Okay. Thank you for clarifying.
So do you remember talking about how Unix applications were separated and could not affect each other in the context of containers?
A. Yes, generally.
Q. What did you mean by the part of your answer that applications running Unix were not separated and could affect each other?
A. Without any containers, applications running in Unix can affect other applications by the -- well, one of the simplest examples is Application A comes up and uses the system IP address and binds Port No. 1000 to that. Application B comes up, wants to do the same thing and is not able to -- it will receive an error and will crash because that IP number is port number is already in use.
So you've got Unix Application A causing application Unix Application B not to be able to proceed because of the way it uses

networking. It's a very simple example.

Q. And --

A. That would be one Unix behavior, you know.

Q. And is that conflict of port number something that was solved by the technology in either the '814 patent or the '058 patent?

A. Yes.

MR. ANAPOL: Objection. Vague as to the patent and vague as to the technology.

THE WITNESS: Sorry.

BY MR. TONG:

Q. Which of those patents solved this type of port conflict number?

A. It's described in the '814 patent.

Q. When you spoke with Mr. Anapol, did he inform you that Amazon was a defendant in this case, accused of patent infringement?

A. Actually, I don't recall there being a discussion of that.

It was very clear to me from the summons I received that that was the case. I

don't recall talking with Mr. Anapol about anything like that.

Q. Did Mr. Anapol tell you that Amazon intends to pursue a defense in this case that your patents are, in fact, invalid?

A. No, that was not described to me.

Q. Do you think that either of your '814 or '058 patent are invalid?

A. That would not be for me to comment. I'm not a -- I'm -- I'm not -- I have nothing to do with the patent, you know, organization. I'm not a reviewer. I...

Q. You are an inventor, though; right?

A. I am an inventor, yes.

Q. And at the time that you filed for your patent applications, for both of those, you thought that they were, in fact, true inventions in the sense of the -- in the common of the word "invention"; correct?

MR. ANAPOL: Objection. Leading. Objection. Vague. Objection to the extent it calls for a legal conclusion.

THE WITNESS: Obviously we -- when we submitted the patent, we thought that it was inventive, or we wouldn't have submitted it.

BY MR. TONG:

Q. And has your opinion changed since the day you had filed the application?

MR. ANAPOL: Objection. Legal conclusion. Lacks foundation.

THE WITNESS: Well --

MR. ANAPOL: And to the extent it calls for expert opinion.

THE WITNESS: Yeah, I mean, it -- I'd have to -- I can only give you an opinion.

I mean, there are -- at the time the biggest advantage was use of this kind of containerization in Windows and open source Linux kinds of things because it wasn't available at the time. Today there are lots of better solutions.

BY MR. TONG:

Q. If Amazon were to attempt to invalidate your patents at trial and to argue that what you

filed for is not really an invention, would you be willing to show up and testify on the stand and defend the validity and the integrity of your status as an inventor?

A. Does it change the status of my -- no. The answer is no. I don't have -- I do not feel the need to defend the work we did or defend this patent. I don't feel the need to do that, and I would not do that unless I wasn't given any option.

Q. Okay. And is that because you're comfortable knowing that you are, in fact, an inventor and there's no one you need to prove that to?

MR. ANAPOL: Objection. Leading.

THE WITNESS: Yeah, I don't need to prove myself to anybody. I'm very comfortable with what I've accomplished and what these things -- they are what they are, and...

MR. TONG: And could we -- we've been going for a little bit could we take a short maybe five- or ten-minute break, and then I'll

wrap up?

THE WITNESS: Yeah, fine by me.

MR. TONG: Okay. Thanks.

MR. ANAPOL: Let's go off the record, and then I want to talk about the break.

THE VIDEOGRAPHER: Going off the record. The time is 1:58.

(Recess taken.)

THE VIDEOGRAPHER: We are back on the record. The time is 2:06.

BY MR. TONG:

Q. Thank you, Mr. Rochette. I have just five more lines of questioning. They should hopefully all be short.

The first one, again, relates to your willingness to attend trial. It's possible that Amazon may raise a defense and say that you had, in fact, committed fraud upon the patent office because you should have told the patent office about your work at OnCore or your knowledge of Solaris zones.

If Amazon tried to accuse you of fraud

at this trial, would you in that case be willing to show up and defend yourself?

MR. ANAPOL: Objection. Lacks foundation. Amazon hasn't done that. Calls for speculation.

THE WITNESS: No. I have no -- they can -- once again, I don't feel the need to defend myself. It is what it is. I -- and...

BY MR. TONG:

Q. Thanks.

Are you proud of being an inventor on these patents?

A. I feel like we did good work, yeah, that was needed at the time.

Q. For --

MR. ANAPOL: Outside the -- sorry.

Outside the scope.

BY MR. TONG:

Q. Are you a lawyer or do you have any type of law degree?

A. No.

Q. Are you aware of any orders from the

court designating you as an expert in this case?

A. No.

Q. Can you confirm that you have not received any claim constructions for either the '814 or '058 patent from the court?

A. What did you describe it as? Claim what?

Q. Claim constructions.

I take it from your question that you haven't received any from the court; is that correct?

A. No. I don't even know what it is.

Q. From the '814 patent, which is titled "System for Containerization of Application Sets," could you please look into the camera, and -- assuming that this may get shown to the judge or jury if we're unable to call you at trial because you're not willing to attend, and if this is the only thing that the jury hears from you about the '814 patent, what would you say to the jury about why the '814 patent was a new and inventive technology at the time that

you had filed it?

MR. ANAPOL: Objection. Outside the scope. Lacks foundation. Assumes facts not in evidence and calls for speculation.

THE WITNESS: At the time that this was -- these applications were submitted, there was no generally accepted definition of what containers are. You have to put yourself back in the -- back in time. Today containers are well understood and are everywhere, and at the time there was nothing like that.

Very close to the time that this came out -- again, as I've said many times, I don't know the exact timing -- Solaris had a version that did something similar that was specific to the Solaris operating system. You could not do what this -- these patents describe on Linux or Unix or Windows. And so in particular on Windows, it was a -- it was a solution. It was a way to create a form of container capability that was not present at the time.

So if you look at things now, you

can -- you know, it's a very different world, but at the time, particularly on Windows, you couldn't do anything like this.

And one of the things that was thought to be quite useful and inventive was an ability to run older versions of an application on newer versions of an operating system. So companies could upgrade their systems, get requisite security updates and stuff that came with newer operating systems and also be able to continue running those old applications without having to rebuild and recertify, all kinds of things like that. It was -- for Windows and Unix, Linux, it was helpful and useful.

BY MR. TONG:

Q. Thank you.

And you already somewhat started to answer my next question, but, again, if you just had one chance to say something to the judge or jury about why the invention in the '814 patent was valuable and worth patenting, what would you say to the jury?

MR. ANAPOL: Same objections. And vague as to "value."

THE WITNESS: Well, again I would say that at the time that the provisionals were submitted, there wasn't the capability to do this in any way on Windows. Now, there might have been something that you could play with on Linux or something like that, but it wasn't -- it wasn't a full-feature product or capability, and the use of interposition/function overlays in order to accomplish this was quite unique, quite inventive.

As I outlined the discussion with the guys at Solaris, they didn't think it was possible. We sat down and went through, here's how we did it, and there's a whole community of people that went, oh, yeah, that's a -- that is different, that's unique, that's inventive and, you know, works well with Unix and Windows. So I would say that at the time it was a valuable way to consider container techniques and capabilities.

I'll also reiterate that it really had nothing to do with OnCore. What OnCore was doing was very different. Despite the broad use of the term "container," they're not the same.

BY MR. TONG:

Q. Okay. Thank you.

Next two questions are going to be about the '058 patent.

So for the '058 patent, which is titled "Computing System Having User Mode Critical System Elements As Shared Libraries," if you could only have one chance to say something to the jury about why the invention in this patent is, in fact, new and inventive, what would you tell the jury?

MR. ANAPOL: Same objections.

THE WITNESS: It was taking functionality that heretofore at the time of the invention was only available and only implemented in protected kernel-space code. It wasn't moving it out of protected space. It wasn't copying it. It was performing

similar-type behaviors in user mode. That's the term "critical system elements."

The -- maybe the simplest example is the use of Application A in one of these container environments could use an IP address; Application B could use a different IP address. With -- you could not do that without the capability in these function overlays and this interposition capability. The applications would not function without an ability to manage these networking components.

Same example with versions of the operating system; right? An older version of an -- an application that was running on an older version of the operating system could not, at the time that this was created, run on a newer version of the operating system because the libraries and other configuration files were not compatible. And so you -- this gave you the ability to maintain compatibility.

The big -- the important thing about this is it gave people in the data center the

ability to upgrade their operating system without having to go back and recompile, recertify, rebuild, redeploy certain applications.

Q. Thank you.

And last question: For the '058 patent at the time of filing, if you had one chance to explain to the judge and jury why the invention was valuable, what would you tell them?

MR. ANAPOL: Same objections.

THE WITNESS: I think I would focus on the -- 1, the use of this in Windows. At the time this was filed, you could not do anything like this on Windows.

No. 2, the ability to run older applications on new operating systems. At that time it was a pretty big deal. It's a very big deal. And if you think about, for example, security updates -- everybody knows about security updates now. It's very important to be able to get a new version of the operating system apply updates to that which include all

the security updates and stuff you need. When you do that, you break the older applications.

So what is your option? Your option is to run with an older version of the operating system, to pay more licensing fees, to pay -- to run with relatively less secure environments.

This technology, as defined in '058, would allow you to update those environments, provide consistency across your data center, get into a more secure environment, manage the number of licenses you have, and at the same time be able to continue to use the applications that have been around for a while.

That is one of the major advantages that people saw with the technology as defined in '058.

BY MR. TONG:

Q. Thanks. Just one follow-up question.

You said that was one of the major advantages.

Were there any other secondary or tertiary advantages?

MR. ANAPOL: Same objections.

THE WITNESS: The ability to get around the port number limitations was another -- was an advantage. The ability to use multiple IP addresses on the same environment that would otherwise be rather difficult to do are a couple of other examples.

MR. TONG: All right. Thank you so much for your time, Mr. Rochette.

Jeremy, I pass the witness.

MR. ANAPOL: Thank you, Peter.

FURTHER EXAMINATION

BY MR. ANAPOL:

Q. So, Mr. Rochette, I just have a few follow-up questions related to what Mr. Tong asked you about.

So he mentioned your conversation with Nigel Stokes.

Do you remember that?

A. Yes.

Q. When you worked at Trigence and

developed the '058 and '814 patents, was Mr. Stokes there?

A. Not to my knowledge, no. Never spoke with -- no.

Q. So when did Mr. Stokes become interested in these patents?

A. I don't know when. I just know that a couple months ago he reached out to me on LinkedIn.

Q. So Mr. Stokes had no interest in the patents at the time they were filed; correct?

A. That's my understanding.

MR. TONG: Objection. Calls for speculation.

BY MR. ANAPOL:

Q. All right. Okay. And we talked earlier about the patent claims versus the rest of the patent.

Do you remember that?

A. Yes.

Q. Did you write the claims in either the '814 or the '058 patent?

A. They were written by a patent attorney, not by me.

Q. And the testimony you gave today about differences between the patents and other technology we discussed, was that informed by the claims?

A. No.

Q. So it's possible that the lawyer who wrote the claims didn't capture the distinctions that you discussed; is that right?

A. I could only speculate, but -- yeah, I don't know.

Q. Well, we don't want you to speculate, so let's talk about one of the things you identified.

A. Okay.

Q. You said that before the patents, there was some container capability on Solaris but not on Windows.

Do you remember that?

A. Yes.

MR. ANAPOL: Billy, can you pull up

Exhibit 1010 and go to the claims, which are at the end of the patent, starting Column 17. And can you blow up Claim 1 for us.

Q. So, Mr. Rochette, do you see anything in the claim that requires the invention to be implemented on Windows or Linux?

MR. TONG: Objection. Calls for a legal conclusion, expert opinion. Daubert. Also outside the scope of venue discovery.

THE WITNESS: I don't think I've -- no. And I don't think I've represented that it required -- that anything in the patent is required to run on Windows.

What I feel like I've been describing is it is able to run on Windows, and at the time that was a big enabler. It was also able to run on Windows and Linux.

BY MR. ANAPOL:

Q. And it was able to run on Solaris; right?

A. It was able to run on Solaris.

Q. And nothing in the patent claim says

we're only claiming the invention outside of Solaris; right?

MR. TONG: Objection. Foundation. Scope. Calls for a legal conclusion, expert opinion. Daubert.

THE WITNESS: To my understanding, there's nothing in this patent that says that it is limited to Solaris or -- that it's limited to any specific operating system. I don't -- that's not my understanding.

BY MR. ANAPOL:

Q. Okay. Can you point to any language in this claim that reflects the idea that you are the first to do this type of container functionality on Windows or Linux?

MR. TONG: Objection. Foundation. Scope. Calls for legal conclusions, expert opinion. Daubert.

MR. ANAPOL: Okay. I'll withdraw the question.

Q. Did you look at the patent claims and compare them to Solaris zones or the OnCore

operating system?

A. No. Not in preparation for this and -- no.

Q. Did --

A. I didn't look at the claims. I didn't read the claims.

Q. Did you ever have a conversation with the patent attorney who wrote the claims to explain how OnCore operating system worked?

MR. TONG: Objection. Potentially privileged.

Mr. Rochette, if you think your conversations with your attorneys are privileged, you absolutely do not have to answer this question.

THE WITNESS: I don't --

MR. TONG: And, in fact, if you did this as part of your work for either AppZero or Trigence, you may not be able to waive privilege at all. So I would suggest that you do not answer this question.

THE WITNESS: Okay.

BY MR. ANAPOL:

Q. Do you recall ever providing any documentation about the OnCore operating system to the patent attorney who answered these claims?

MR. TONG: Same objection. You don't have to answer this question.

Privileged.

BY MR. ANAPOL:

Q. You're free, Mr. Rochette, to, I guess, make your own decision here. Mr. Tong doesn't represent you.

I'll caution you that to the extent you don't answer a question because Mr. Tong thinks it's privileged and later we show that it's not privileged, then we might have to have another deposition to have you answer the question.

A. I had no conversations with anything -- about OnCore. No conversations about OnCore. It is my understanding and belief that OnCore is very different than and unrelated to these patents.

Q. To the patent specification or to the claims?

A. Both.

Q. And how did you reach your conclusion that OnCore is different from the claims?

A. OnCore is doing something extremely different. OnCore has nothing do with any function overlays, any preloading, any networking redirection. OnCore didn't do any of that at all.

Q. So you just listed three things: function overlays, preloading and networking redirections.

Can you point to me in Claim 1 where any of those three concepts are required?

MR. TONG: Objection. Foundation. Calls for a legal conclusion. Calls for expert opinion. Daubert.

THE WITNESS: Again, I haven't read the claims. I don't -- I don't feel capable representing the claims. I can tell you how the technology works. I can try to answer questions

about how it works and what the intent is and things like that, but I don't -- I have -- I don't know what the claims are.

BY MR. ANAPOL:

Q. Okay. But you just testified that OnCore is different from the claims, so I'm just trying to understand the basis for that testimony.

A. Okay.

Q. But if you don't -- if that's not your opinion, then that's fine. I'm just trying to clarify.

MR. TONG: Objection --

THE WITNESS: OnCore is --

MR. TONG: -- asked and answered.

THE WITNESS: OnCore is very different from any of this, from what's described in these -- for the way in which these patents work, OnCore is very different. It is -- it's very, very different.

BY MR. ANAPOL:

Q. And when you say "the way in which

these patents work," are you talking about what's claimed or what is disclosed in the patents?

A. What is disclosed in the patents. Again, I don't -- did not read the claims. I can't represent to you anything about the claims. I don't -- I didn't -- I don't -- I don't know about the claims. I didn't read them.

Q. And your understanding is that those were written by an attorney and not by you; correct?

A. That's true.

MR. TONG: Objection. Asked and answered.

BY MR. ANAPOL:

Q. And your answer that you just gave about the claims and the fact that you haven't read them and didn't write them, is that also true of the '058 patent?

A. Yes.

Q. So whatever distinctions you pointed

out between the '058 patent and either OnCore or Solaris zones or other technology, that was by comparison to the disclosure and not the claims?

MR. TONG: Objection. Mischaracterizes.

THE WITNESS: The implementation -- if by "disclosure" you meant the implementation, then, yes, the way in which -- what is -- yes, the way they're implemented. I didn't represent -- it was not my intention to represent anything in the claims.

BY MR. ANAPOL:

Q. Okay. And, to your knowledge, did you provide any information about the OnCore operating system or the Solaris zones to the patent examiner who was responsible for examining these claims?

A. I'm pretty confident in saying nothing about -- that I didn't -- that there was no conversation or interaction about OnCore. Solaris zones, I don't know. Could have been. I don't recall.

Q. Okay. You don't recall providing any information about Solaris zones to the patent examiner?

A. I do not recall.

MR. TONG: Objection. Asked and answered.

BY MR. ANAPOL:

Q. And were you aware -- actually, I'll withdraw that question. You already answered it.

And were you aware that there was a product called Linux V-Server that provided containers on Linux?

MR. TONG: Objection. Vague. Ambiguous. Foundation.

THE WITNESS: I'm not aware of Linux V-Server, no.

BY MR. ANAPOL:

Q. So when you testified earlier that Linux didn't have the capability provided by these patents, you had not done any comparison between the patents and Linux V-Server; correct?

A. I don't recall doing any -- I don't recall there being any mention of V-Server, no.

Q. And have you ever heard of something called User Mode Linux?

A. Yes.

Q. And what's your understanding of User Mode Linux?

A. I don't recall the details of it. There's -- there's a lot of -- I don't know. I'd have to go back and look at that. I remember -- I remember it, and there were discussions about it, but I don't recall at this moment how it works.

Q. So do you recall ever providing any information about User Mode Linux to the patent examiner?

A. I don't recall answering any questions about that in the context of the patent, no. Not to my recollection.

Q. And so when you testified earlier that certain aspects of the patent seemed inventive to you, did you check to see whether any of

those aspects were in User Mode Linux?

A. I believe there was some -- that we did look at User Mode Linux, but it's not -- the conclusion -- it was pretty easy to conclude that they weren't the same and they weren't going to be -- we were providing something very different from that, but I don't recall the details.

Q. So you can't recall what you thought was inventive about the patents compared to User Mode Linux?

A. No, I can't recall how User Mode Linux works. No. I would have to go back --

MR. TONG: Jeremy --

BY MR. ANAPOL:

Q. Do you --

MR. TONG: -- [overspoken] User Mode Linux and the --

(The reporter requested clarification.)

MR. TONG: Yeah. I'm objecting to the line of questions of User Mode Linux as outside the scope.

MR. ANAPOL: All right. Well, I don't have any more questions on User Mode Linux, so that's fine.

Q. Do you recall testifying earlier, Mr. Rochette, about the capability of running different operating systems -- sorry. Let me rephrase.

Do you recall testifying earlier about running applications from an earlier version of an operating system on a later version of an operating system?

A. Yes.

Q. And do you recall Solaris having that capability?

A. I don't -- I don't recall whether or not you could run older versions of applications in Solaris zones and stuff like that. I don't recall.

Q. You don't know whether Solaris 9 applications could run in zones on Solaris 10?

A. I don't recall, no.

Q. And are you aware of a system called

Alpine from the University of Washington?

MR. TONG: Objection to the scope of questioning as outside the scope of cross.

THE WITNESS: I am aware of Alpine Linux.

BY MR. ANAPOL:

Q. Oh, sorry. Not Alpine Linux. It's a library called Alpine that ran on top of BSD.

A. Oh. No, I'm not aware of that.

MR. TONG: Same objection. Outside the scope.

BY MR. ANAPOL:

Q. So when you testified earlier about having this ability to avoid port conflicts as part of the patents, you don't know whether the Alpine library that ran on BSD had that capability or not?

A. I'm not aware --

MR. TONG: Same objection.

THE WITNESS: -- of Alpine, no. I don't know how it works.

BY MR. ANAPOL:

Q. And we talked a lot about LD_Preload and function overlays.

Do you remember that?

A. Yes.

Q. And do you know if the function overlay aspect of what you developed was recited in the patent claims?

A. The behavior was described, yes -- oh, in the claims. You asked about the claims. No, I don't know about the claims. No.

MR. ANAPOL: Thank you so much for your time, Mr. Rochette. No further questions.

FURTHER EXAMINATION

BY MR. TONG:

Q. I have just one very short line of questioning, Mr. Rochette, and then we should be done after this.

So when Jeremy was questioning you, you had answered that you have not read or written the claims.

When you had filed for the -- strike that.

When you were in the process of filing for the '058 patent application, would it have been your intent for the claims to capture the technology as you understood and described it today?

MR. ANAPOL: Objection. Vague. Lacks foundation. Calls for speculation. Improper hypothetical.

THE WITNESS: That seems logical.

BY MR. TONG:

Q. And same question for the other patent, the -- sorry. I'm blanking on the number.

Same question for the '814 patent: When you were in the process of filing for the '814 patent, would it have been your intent for the claims to capture the technology as you understood and described it today?

MR. ANAPOL: Same objections.

THE WITNESS: I don't -- I honestly don't -- I would -- I don't know. The claims

were very -- very abstract to me. The lawyer did them, and we had conversations about the technology and how it worked. I would --

BY MR. TONG:

Q. Don't --

MR. ANAPOL: Don't --

BY MR. TONG:

Q. Don't discuss anything --

MR. ANAPOL: Peter -- Peter, you have to let him finish his answer.

MR. TONG: Well, I'm going to tell him not to reveal anything privileged.

Q. So to the extent you're about to say anything about your discussions with a lawyer, don't reveal that. But otherwise, go ahead and finish answering the question.

A. Yeah, in a very -- discussions about the claims were always very vague and very legal, and I would -- I don't know. I make no -- I don't know how to -- I don't know how to defend or answer your questions about claims. I just don't. I'm not the right guy to do that.

Q. Yeah, to clarify my question, it's not whether or not the claims ended up one way or another. My question was about your intent at the time.

Would it have been your intent for your patent to capture the technology as you understood and described it today?

MR. ANAPOL: Same objections --

THE WITNESS: It was my --

MR. ANAPOL: -- and relevance.

THE WITNESS: It was my intent to get the description of the invention -- to describe it as it worked. That was my intent. It's up to the company and the lawyers and so forth to nail down the claims. That was not my -- I wasn't involved. Not my -- no intent. My intent was to be able to describe how the invention worked.

MR. TONG: Okay. Thank you, Mr. Rochette.

Pass the witness. And we should be done.

Can you confirm, Jeremy?

MR. ANAPOL: Yes.

Thanks again for your time, Mr. Rochette.

MR. TONG: Thank you so much.

THE WITNESS: Am I free to go?

MR. ANAPOL: You are free to go.

MR. TONG: Mr. Rochette -- we can go off the record.

THE VIDEOGRAPHER: Okay. We are going off the record. This deposition is concluded at 2:39.

(Discussion held off the record.)

THE REPORTER: Counsel, do you need a rough draft of this?

MR. ANAPOL: Yes, please.

THE REPORTER: And, Peter, do you need a rough?

MR. TONG: Yes.

(Whereupon, at 2:44 P.M. CDT the deposition of DONN ROCHETTE was adjourned.)

STATE OF CALIFORNIA)
COUNTY OF LOS ANGELES) SS.

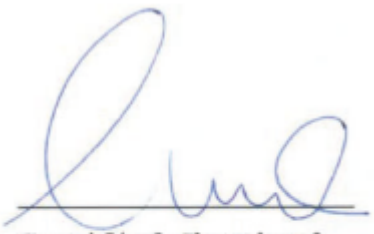
I, AUDRA E. CRAMER, CSR No. 9901, in and for the State of California, do hereby certify:

That, prior to being examined, the witness named in the foregoing deposition was by me duly sworn to testify the truth, the whole truth and nothing but the truth;

That said deposition was taken down by me in shorthand at the time and place therein named, and thereafter reduced to typewriting under my direction, and the same is a true, correct and complete transcript of said proceedings;

I further certify that I am not interested in the event of the action.

Witness my hand this 18 day of September, 2024.



Certified Shorthand
Reporter for the
State of California

Donn Rochette, c/o
Pro se

Case: VirtaMove Corp. v. Amazon.com, Inc., et al.
Date of deposition: September 10, 2024
Deponent: Donn Rochette

Please be advised that the transcript in the above referenced matter is now complete and ready for signature. The deponent may come to this office to sign the transcript, a copy may be purchased for the witness to review and sign, or the deponent and/or counsel may waive the option of signing. Please advise us of the option selected. Please forward the errata sheet and the original signed signature page to counsel noticing the deposition, noting the applicable time period allowed for such by the governing Rules of Procedure. If you have any questions, please do not hesitate to call our office at (202)-232-0646.

Sincerely,
Digital Evidence Group

Digital Evidence Group, L.L.C.
1730 M Street, NW, Suite 812
Washington, D.C. 20036
(202) 232-0646

SIGNATURE PAGE
Case: VirtaMove Corp. v. Amazon.com, Inc., et al.
Witness Name: Donn Rochette
Deposition Date: September 10, 2024

I do hereby acknowledge that I have read and examined the foregoing pages of the transcript of my deposition and that:

(Check appropriate box):
( ) The same is a true, correct and complete transcription of the answers given by me to the questions therein recorded.
( ) Except for the changes noted in the attached Errata Sheet, the same is a true, correct and complete transcription of the answers given by me to the questions therein recorded.

_____ _____
DATE WITNESS SIGNATURE

_____ _____
DATE NOTARY

Digital Evidence Group, LLC
1730 M Street, NW, Suite 812
Washington, D.C. 20036
(202)232-0646

ERRATA SHEET

Case: VirtaMove Corp. v. Amazon.com, Inc., et al.
Witness Name: Donn Rochette
Deposition Date: September 10, 2024

Page No.	Line No.	Change

_____ _____
Signature Date

S